9.1

Sécurisation d' IBM MQ

Remarque

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Remarques», à la page 677.

Cette édition s'applique à la version 9 édition 1 d' IBM® MQ et à toutes les éditions et modifications ultérieures, sauf indication contraire dans les nouvelles éditions.

Table des matières

Sécurisation de IBM MQ

La sécurité est une considération importante pour les développeurs d'applications IBM MQ et pour les administrateurs système IBM MQ .

Mises à jour de sécurité

Vérifiez que tous les matériels et logiciels à l'intérieur de la zone sécurisée et sur les postes de travail de l'opérateur sont dans leur cycle de vie de support, ont été mis à niveau avec des mises à jour logicielles obligatoires et ont fait l'objet de mises à jour de sécurité rapidement appliquées.

Vous trouverez plus d'informations sur les mises à jour de sécurité pour:

- Toutes les plateformes sur le site IBM Security Bulletins
- APAR de sécurité et d'intégrité du système sur z/OS sur le portailIBM Z System Integrity.

Présentation de la sécurité

Cette collection de rubriques présente les concepts de sécurité d' IBM MQ .

Les concepts et les mécanismes de sécurité, tels qu'ils s'appliquent à n'importe quel système informatique, sont présentés en premier, suivis d'une discussion de ces mécanismes de sécurité à mesure qu'ils sont implémentés dans IBM MQ.

Concepts et mécanismes de sécurité

Cette collection de rubriques décrit les aspects de la sécurité à prendre en compte dans votre installation IBM MQ .

Les aspects communément acceptés de la sécurité sont les suivants:

- «Identification et authentification», à la page 6
- «Authorization», à la page 6
- «Audit», à la page 7
- «Confidentialité», à la page 7
- «Intégrité des données», à la page 7

Les *mécanismes de sécurité* sont des outils et des techniques techniques utilisés pour implémenter les services de sécurité. Un mécanisme peut fonctionner seul ou avec d'autres pour fournir un service particulier. Voici des exemples de mécanismes de sécurité communs:

- «Cryptographie», à la page 7
- «Historiques des messages et signatures numériques», à la page 9
- «Certificats numériques», à la page 9
- «Public key infrastructure (PKI)», à la page 14

Lorsque vous planifiez une implémentation IBM MQ , prenez en compte les mécanismes de sécurité dont vous avez besoin pour implémenter les aspects de la sécurité qui sont importants pour vous. Pour plus d'informations sur les éléments à prendre en compte après avoir lu ces rubriques, voir «Planification de la sécurité», à la page 81.

Concepts associés

«Utilisation de SSL/TLS», à la page 282
Ces rubriques fournissent des instructions pour l'exécution de tâches uniques liées à l'utilisation de TLS avec IBM MQ.

Tâches associées

Connexion de deux gestionnaires de files d'attente via le protocole TLS

Identification et authentification

L' *identification* est la possibilité d'identifier de manière unique un utilisateur d'un système ou d'une application qui s'exécute dans le système. L' *authentification* est la possibilité de prouver qu'un utilisateur ou une application est réellement qui est cette personne ou ce que cette application prétend être.

Par exemple, imaginez un utilisateur qui se connecte à un système en entrant un ID utilisateur et un mot de passe. Le système utilise l'ID utilisateur pour identifier l'utilisateur. Le système authentifie l'utilisateur au moment de la connexion en vérifiant que le mot de passe fourni est correct.

Non-répudiation

Le service de *non-répudiation* peut être considéré comme une extension du service d'identification et d'authentification. En général, la non-répudiation s'applique lorsque les données sont transmises par voie électronique ; par exemple, une commande à un courtier en actions pour acheter ou vendre des actions, ou une commande à une banque pour transférer des fonds d'un compte à un autre.

L'objectif global du service de non-répudiation est de pouvoir prouver qu'un message particulier est associé à un individu particulier.

Le service de non-répudiation peut contenir plusieurs composants, chaque composant fournissant une fonction différente. Si l'expéditeur d'un message refuse de l'envoyer, le service de non-répudiation avec une *preuve de l'origine* peut fournir au destinataire des preuves indéniables que le message a été envoyé par cette personne. Si le destinataire d'un message refuse de le recevoir, le service de non-répudiation avec *preuve de livraison* peut fournir à l'expéditeur des preuves indéniables que le message a été reçu par cette personne.

Dans la pratique, la preuve avec une quasi-certitude de 100%, ou des preuves indéniables, est un objectif difficile. Dans le monde réel, rien n'est totalement sécurisé. La gestion de la sécurité est plus axée sur la gestion des risques à un niveau acceptable pour l'entreprise. Dans un tel environnement, une attente plus réaliste du service de non-répudiation est de pouvoir fournir des preuves qui sont admissibles, et qui soutiennent votre cause, devant un tribunal.

La non-répudiation est un service de sécurité pertinent dans un environnement IBM MQ car IBM MQ est un moyen de transmettre des données par voie électronique. Par exemple, vous pouvez exiger des preuves simultanées qu'un message particulier a été envoyé ou reçu par une demande associée à un individu particulier.

IBM MQ avec Advanced Message Security ne fournit pas de service de non-répudiation dans le cadre de sa fonction de base. Toutefois, cette documentation produit contient des suggestions sur la manière dont vous pouvez fournir votre propre service de non-répudiation dans un environnement IBM MQ en écrivant vos propres programmes d'exit.

Concepts associés

«Identification et authentification dans IBM MQ», à la page 21
Dans IBM MQ, vous pouvez implémenter l'identification et l'authentification à l'aide des informations de contexte de message et de l'authentification mutuelle.

Authorization

L' *autorisation* protège les ressources critiques d'un système en limitant l'accès uniquement aux utilisateurs autorisés et à leurs applications. Il empêche l'utilisation non autorisée d'une ressource ou l'utilisation d'une ressource de manière non autorisée.

Concepts associés

«Autorisation dans IBM MQ», à la page 22
Vous pouvez utiliser l'autorisation pour limiter les actions que peuvent effectuer des individus ou des applications spécifiques dans votre environnement IBM MQ .

Audit

L' *audit* est le processus d'enregistrement et de vérification des événements pour détecter si une activité inattendue ou non autorisée a eu lieu ou si une tentative a été effectuée pour effectuer cette activité.

Pour plus d'informations sur la configuration de l'autorisation, voir «Autorisation de planification», à la page 85 et les sous-rubriques associées.

Concepts associés

«Audit dans IBM MQ», à la page 22
IBM MQ peut émettre des messages d'événement pour enregistrer que l'activité inhabituelle a eu lieu.

Confidentialité

Le service de *confidentialité* protège les informations sensibles contre toute divulgation non autorisée.

Lorsque des données sensibles sont stockées localement, les mécanismes de contrôle d'accès peuvent être suffisants pour les protéger en supposant que les données ne peuvent pas être lues si elles ne sont pas accessibles. Si un niveau de sécurité plus élevé est requis, les données peuvent être chiffrées.

Chiffrer des données sensibles lorsqu'elles sont transmises sur un réseau de communication, en particulier sur un réseau non sécurisé tel que l'Internet. Dans un environnement réseau, les mécanismes de contrôle d'accès ne sont pas efficaces contre les tentatives d'interception des données, telles que les écoutes téléphoniques.

Intégrité des données

Le service d' *intégrité des données* détecte s'il y a eu une modification non autorisée des données.

Les données peuvent être altérées de deux manières: accidentellement, par des erreurs de matériel et de transmission, ou en raison d'une attaque délibérée. De nombreux produits matériels et protocoles de transmission ont des mécanismes pour détecter et corriger les erreurs matérielles et de transmission. Le but du service d'intégrité des données est de détecter une attaque délibérée.

Le service d'intégrité des données vise uniquement à détecter si des données ont été modifiées. Il ne vise pas à restaurer les données dans leur état d'origine si elles ont été modifiées.

Les mécanismes de contrôle d'accès peuvent contribuer à l'intégrité des données dans la mesure où les données ne peuvent pas être modifiées si l'accès est refusé. Mais, comme pour la confidentialité, les mécanismes de contrôle de l'accès ne sont pas efficaces dans un environnement de réseautage.

Concepts cryptographiques

Cette collection de rubriques décrit les concepts de cryptographie applicables à IBM MQ.

Le terme *entité* est utilisé pour désigner un gestionnaire de files d'attente, un IBM MQ MQI client, un utilisateur individuel ou tout autre système capable d'échanger des messages.

Concepts associés

«Cryptographie dans IBM MQ», à la page 23
IBM MQ fournit la cryptographie à l'aide du protocole TLS (Transport Security Layer).

Cryptographie

La cryptographie est le processus de conversion entre du texte lisible, appelé *texte en clair*, et un format illisible, appelé *texte chiffré*.

Cela se produit comme suit:

1. L'expéditeur convertit le message en texte en clair en texte chiffré. Cette partie du processus est appelée *chiffrement* (parfois *chiffrement*).
2. Le texte chiffré est transmis au récepteur.
3. Le récepteur reconvertit le message chiffré en texte en clair. Cette partie du processus est appelée *déchiffrement* (parfois *déchiffrement*).

La conversion implique une séquence d'opérations mathématiques qui modifient l'apparence du message lors de la transmission mais n'affectent pas le contenu. Les techniques cryptographiques permettent de garantir la confidentialité et de protéger les messages contre l'affichage non autorisé (écoute clandestine), car un message chiffré n'est pas compréhensible. Les signatures numériques, qui garantissent l'intégrité des messages, utilisent des techniques de chiffrement. Pour plus d'informations, voir «Signatures numériques dans SSL/TLS», à la page 19.

Les techniques cryptographiques impliquent un algorithme général, rendu spécifique par l'utilisation de clés. Il existe deux classes d'algorithme:

- Ceux qui exigent que les deux parties utilisent la même clé secrète. Les algorithmes qui utilisent une clé partagée sont appelés algorithmes *symétriques* . La Figure 1, à la page 8 illustre la cryptographie à clé symétrique.
- Ceux qui utilisent une clé pour le chiffrement et une autre clé pour le déchiffrement. L'un d'eux doit être gardé secret, mais l'autre peut être public. Les algorithmes qui utilisent des paires de clés publiques et privées sont appelés algorithmes *asymétriques* . La Figure 2, à la page 8 illustre la cryptographie à clé asymétrique, également appelée *cryptographie à clé publique*.

Les algorithmes de chiffrement et de déchiffrement utilisés peuvent être publics, mais la clé secrète partagée et la clé privée doivent être gardées secrètes.

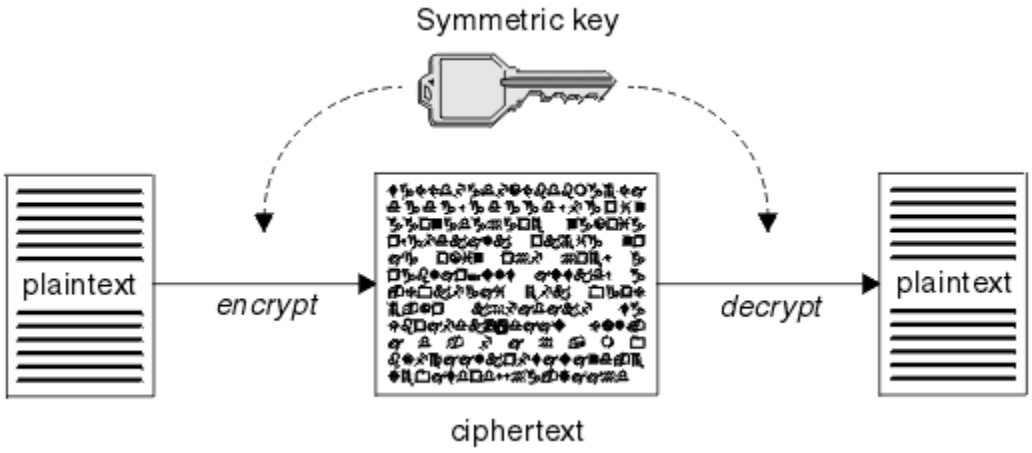


Figure 1. cryptographie à clé symétrique

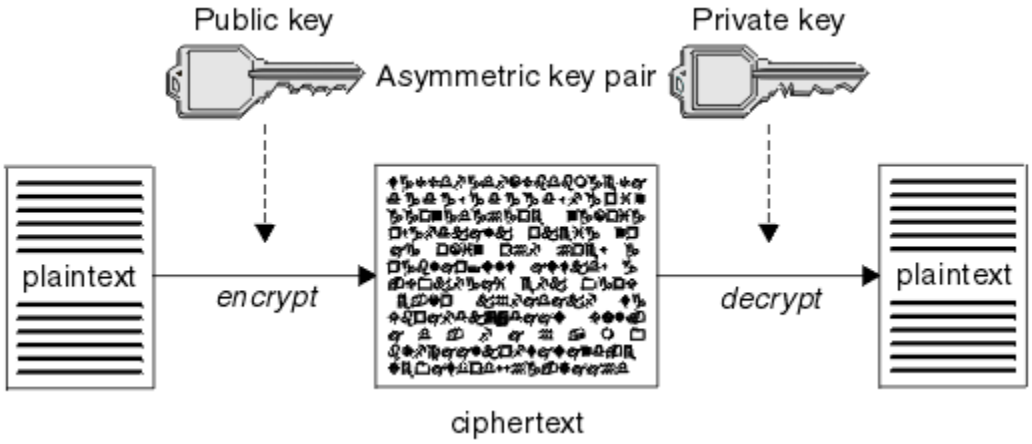


Figure 2. cryptographie à clé asymétrique

Figure 2, à la page 8 affiche le texte en clair chiffré avec la clé publique du récepteur et déchiffré avec la clé privée du récepteur. Seul le récepteur prévu détient la clé privée pour déchiffrer le texte chiffré. Notez que l'expéditeur peut également chiffrer les messages avec une clé privée, ce qui permet à quiconque détient la clé publique de l'expéditeur de déchiffrer le message, avec l'assurance que le message doit provenir de l'expéditeur.

Avec les algorithmes asymétriques, les messages sont chiffrés avec la clé publique ou la clé privée, mais ils ne peuvent être déchiffrés qu'avec l'autre clé. Seule la clé privée est secrète, la clé publique peut être connue de n'importe qui. Avec les algorithmes symétriques, la clé partagée ne doit être connue que des deux parties. Il s'agit du *problème de distribution de clé*. Les algorithmes asymétriques sont plus lents mais présentent l'avantage qu'il n'y a pas de problème de distribution de clé.

Une autre terminologie associée à la cryptographie est:

Force
La force du chiffrement est déterminée par la taille de la clé. Les algorithmes asymétriques requièrent des clés de grande taille, par exemple:

1 024 bits	Clé asymétrique de faible puissance
2048 bits	Clé asymétrique de niveau moyen
4096 bits	Clé asymétrique de haute résistance

Les clés symétriques sont plus petites: les clés 256 bits vous donnent un chiffrement renforcé.

Algorithme de chiffrement de bloc
Ces algorithmes chiffrent les données par blocs. Par exemple, l'algorithme RC2 de RSA Data Security Inc. utilise des blocs d'une longueur de 8 octets. Les algorithmes de bloc sont généralement plus lents que les algorithmes de flux.

Algorithme de chiffrement de flux
Ces algorithmes fonctionnent sur chaque octet de données. Les algorithmes de flux sont généralement plus rapides que les algorithmes de bloc.

Historiques des messages et signatures numériques

Un résumé de message est une représentation numérique de taille fixe du contenu d'un message. Le résumé du message est calculé par une fonction de hachage et peut être chiffré, formant une signature numérique.

La fonction de hachage utilisée pour calculer un résumé de message doit répondre à deux critères:

- Ça doit être un moyen. Il ne doit pas être possible d'inverser la fonction pour trouver le message correspondant à un résumé de message particulier, sauf en testant tous les messages possibles.
- Il doit être infaisable du point de vue du calcul de trouver deux messages qui se hachent dans le même condensé.

Le résumé du message est envoyé avec le message lui-même. Le destinataire peut générer un prétraitement pour le message et le comparer avec le prétraitement de l'expéditeur. L'intégrité du message est vérifiée lorsque les deux historiques de message sont identiques. Toute altération du message au cours de la transmission se traduit presque certainement par un résumé de message différent.

Un résumé de message créé à l'aide d'une clé symétrique secrète est appelé code d'authentification de message (MAC), car il peut fournir l'assurance que le message n'a pas été modifié.

L'expéditeur peut également générer un résumé de message, puis chiffrer le résumé à l'aide de la clé privée d'une paire de clés asymétriques, formant une signature numérique. La signature doit ensuite être déchiffrée par le récepteur, avant de la comparer à un prétraitement généré localement.

Concepts associés

«Signatures numériques dans SSL/TLS», à la page 19
Une signature numérique est formée par le chiffrement d'une représentation d'un message. Le chiffrement utilise la clé privée du signataire et, pour des raisons d'efficacité, opère généralement sur un résumé de message plutôt que sur le message lui-même.

Certificats numériques

Les certificats numériques constituent une protection contre l'usurpation d'identité en certifiant qu'une clé publique appartient à une entité spécifiée. Ils sont émis par une autorité de certification.

Les certificats numériques offrent une protection contre l'emprunt d'identité, car un certificat numérique lie une clé publique à son propriétaire, qu'il s'agisse d'un individu, d'un gestionnaire de files d'attente ou d'une autre entité. Les certificats numériques sont aussi appelés certificats de clé publique car ils donnent des garanties sur l'appartenance d'une clé publique lorsque vous utilisez un schéma de clé asymétrique. Un certificat numérique contient la clé publique d'une entité et établit que la clé publique appartient à cette entité :

- Lorsque le certificat est établi pour une entité individuelle, il est appelé *certificat personnel* ou *certificat d'utilisateur*.
- Lorsque le certificat est établi pour une autorité de certification, il est appelé *certificat d'autorité de certification* ou *certificat de signataire*.

Si les clés publiques sont envoyées directement par leur propriétaire à une autre entité, il existe un risque que le message soit intercepté et que la clé publique soit remplacée par une autre. C'est ce qu'on appelle une attaque de type *homme au milieu*. La solution à ce problème consiste à échanger les clés publiques par le biais d'un tiers sécurisé qui garantit fortement que la clé publique appartient réellement à l'entité avec laquelle vous communiquez. Au lieu d'envoyer votre clé publique directement, vous demandez au tiers sécurisé de l'incorporer dans un certificat numérique. Le tiers sécurisé qui émet les certificats numériques est appelé autorité de certification, comme décrit dans «Autorités de certification», à la page 11.

Qu'est-ce qu'un certificat numérique ?
Les certificats numériques contiennent des éléments spécifiques d'informations, conformément à la norme X.509.

Les certificats numériques utilisés par IBM MQ sont conformes à la norme X.509, qui spécifie les informations requises et le format nécessaire pour leur envoi. X.509 constitue la partie de l'infrastructure d'authentification de la série X.500 de normes.

Les certificats numériques contiennent au moins les informations suivantes sur l'entité qui est certifiée :

- La clé publique du propriétaire
- Le nom distinctif du propriétaire
- Le nom distinctif de l'autorité de certification qui a émis le certificat
- La date à partir de laquelle le certificat est valide
- La date d'expiration du certificat
- Le numéro de version du format de données du certificat, comme défini dans la norme X.509. La version en cours de la norme X.509 est la version 3, et la plupart des certificats sont conformes à cette version.
- Un numéro de série. Il s'agit d'un identificateur unique affecté par l'autorité de certification qui a émis le certificat. Le numéro de série est unique au sein de l'autorité de certification qui a émis le certificat : deux certificats signés par la même autorité de certification ne peuvent pas avoir le même numéro de série.

Un certificat X.509 de version 2 contient aussi un identificateur d'émetteur et un identificateur d'objet, et un certificat X.509 de version 3 peut contenir un certain nombre d'extensions. Certaines extensions de certificat, comme l'extension Contrainte de base, sont *standard*, alors que d'autres sont propres à l'implémentation. Une extension peut être *critique*, auquel cas un système doit pouvoir reconnaître la zone ; s'il ne reconnaît pas la zone, il doit rejeter le certificat. Si une extension n'est pas critique, le système peut l'ignorer s'il ne la reconnaît pas.

La signature numérique dans un certificat personnel est générée avec la clé privée de l'autorité de certification qui a signé ce certificat. Toute personne devant vérifier le certificat personnel peut utiliser la clé publique de l'autorité de certification pour ce faire. Le certificat de l'autorité de certification contient sa clé publique.

Les certificats numériques ne contiennent pas votre clé privée. Vous devez garder votre clé privée secrète.

Exigences relatives aux certificats personnels
IBM MQ prend en charge les certificats numériques conformes à la norme X.509 . Elle requiert l'option d'authentification du client.

IBM MQ étant un système d'égal à égal, il est considéré comme une authentification client dans la terminologie SSL/TLS. Par conséquent, tout certificat personnel utilisé pour l'authentification SSL/TLS doit autoriser une utilisation de clé de l'authentification client. Cette option n'étant pas activée pour tous les certificats serveur, le fournisseur de certificat peut avoir besoin d'activer l'authentification client sur l'autorité de certification racine pour le certificat sécurisé.

En plus des normes qui spécifient le format de données d'un certificat numérique, il existe également des normes pour déterminer si un certificat est valide. Ces normes ont été mises à jour au fil du temps afin de prévenir certains types de violations de la sécurité. Par exemple, les anciens certificats X.509 versions 1 et 2 n'indiquent pas si le certificat peut être légitimement utilisé pour signer d'autres certificats. Il a donc été possible pour un utilisateur malveillant d'obtenir un certificat personnel d'une source légitime et de créer de nouveaux certificats destinés à usurper l'identité d'autres utilisateurs.

Lors de l'utilisation de certificats X.509 version 3, les extensions de certificat BasicConstraints et KeyUsage sont utilisées pour spécifier les certificats qui peuvent légitimement signer d'autres certificats. La norme IETF RFC 5280 spécifie une série de règles de validation de certificat que les logiciels d'application conformes doivent implémenter afin d'éviter les attaques d'usurpation d'identité. Un ensemble de règles de certificat est appelé règle de validation de certificat.

Pour plus d'informations sur les règles de validation de certificat dans IBM MQ, voir «Règles de validation de certificat dans IBM MQ», à la page 44.

Autorités de certification
Une autorité de certification est un tiers sécurisé qui émet des certificats numériques pour garantir que la clé publique d'une entité appartient réellement à cette entité.

Les rôles d'une autorité de certification sont les suivants :

- A la réception d'une demande de certificat numérique, vérifier l'identité du demandeur avant de générer, signer et renvoyer le certificat personnel
- Fournir sa propre clé publique dans son certificat d'autorité de certification
- Publier des listes de certificats qui ne sont plus sécurisés dans une liste de révocation de certificat Pour plus d'informations, voir «Utilisation des certificats révoqués», à la page 348
- Fournir l'accès au statut de révocation de certificat via un serveur répondeur OCSP

Noms distinctifs
Le nom distinctif identifie de façon unique une entité dans un certificat X.509.

Avertissement : Seuls les attributs du tableau suivant peuvent être utilisés dans un filtre SSLPEER. Les noms distinctifs de certificat peuvent contenir d'autres attributs, mais le filtrage n'est pas autorisé sur ces attributs.

Tableau 1. Types d'attribut trouvés dans le nom distinctif pouvant être utilisés dans un filtre SSLPEER

Type d'attribut	**Description**
SERIALNUMBER	Numéro de série du certificat
MAIL	Adresse électronique
E	Adresse électronique (dépréciée dans la préférence pour MAIL)
UID ou USERID	Identificateur utilisateur
CN	Nom CN
T	Titre
OU	Nom d'unité organisationnelle
DC	Composant de domaine
O	Nom de l'organisation

Tableau 1. Types d'attribut trouvés dans le nom distinctif pouvant être utilisés dans un filtre SSLPEER (suite)	
Type d'attribut	**Description**
STREET	Rue/Première ligne d'adresse
L	Nom du lieu
ST (ou SP ou S)	Nom du département
PC	Code postal
C	Pays
UNSTRUCTUREDNAME	Nom d'hôte
UNSTRUCTUREDADDRESS	Adresse IP
DNQ	Qualificateur de nom distinctif

La norme X.509 définit d'autres attributs qui ne font généralement pas partie du nom distinctif mais qui peuvent fournir des extensions en option au certificat numérique.

La norme X.509 permet de spécifier un nom distinctif au format chaîne. Exemple :

```
CN=John Smith, OU=Test, O=IBM, C=GB
```

Le nom usuel (CN) peut décrire un utilisateur individuel ou toute autre entité, par exemple un serveur Web.

Le nom distinctif peut comporter plusieurs attributs OU et DC. Une instance seulement de chacun des autres attributs est admise. L'ordre des entrées OU est important : il spécifie une hiérarchie de noms d'unité organisationnelle, dans laquelle l'unité de niveau supérieur apparaît en premier. L'ordre des entrées DC est également important.

IBM MQ tolère certains noms distinctifs syntaxiquement inappropriés. Pour plus d'informations, voir Règles IBM MQ pour les valeurs SSLPEER.

Concepts associés

«Qu'est-ce qu'un certificat numérique ?», à la page 10
Les certificats numériques contiennent des éléments spécifiques d'informations, conformément à la norme X.509.

Obtention de certificats personnels d'une autorité de certification

Vous pouvez obtenir un certificat d'une autorité de certification externe sécurisée.

Vous obtenez un certificat numérique en envoyant des informations à une autorité de certification, sous la forme d'une demande de certificat. La norme X.509 définit un format pour ces informations, mais certaines autorités de certification proposent leur propre format. Les demandes de certificat sont généralement générées par l'outil de gestion des certificats utilisé par votre système ; par exemple:

- Multi L'outil iKeyman sur Multiplateformes.
- z/OS RACF sur z/OS.

Les informations contiennent votre nom distinctif et votre clé publique. Lorsque votre outil de gestion des certificats génère votre demande de certificat, il génère aussi votre clé privée, qui doit rester sécurisée. Ne la communiquez jamais.

Lorsque l'autorité de certification reçoit votre demande, elle vérifie votre identité avant de générer le certificat et de vous l'envoyer sous forme de certificat personnel.

La Figure 3, à la page 13 illustre le processus d'obtention d'un certificat numérique d'une autorité de certification.

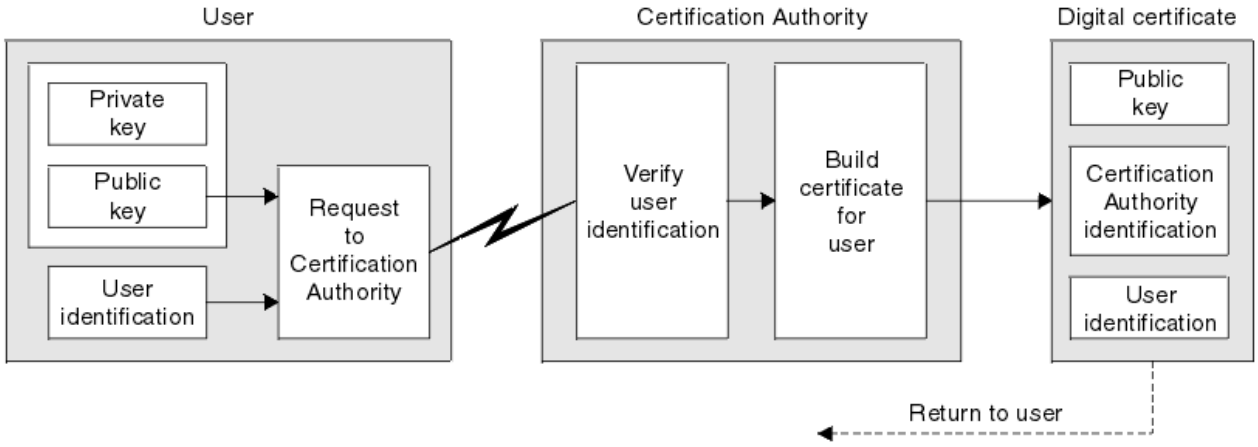


Figure 3. Obtention d'un certificat numérique

Dans le diagramme :

- L'identification de l'utilisateur inclut votre nom distinctif de sujet.
- L'identification de l'autorité de certification inclut le nom distinctif de l'autorité de certification qui émet le certificat.

Les certificats numériques contiennent des zones supplémentaires autres que celles affichées dans le diagramme. Pour plus d'informations sur les autres zones d'un certificat numérique, voir «Qu'est-ce qu'un certificat numérique ?», à la page 10.

Fonctionnement des chaînes de certificats
Lorsque vous recevez le certificat d'une autre entité, vous devrez peut-être utiliser une *chaîne de certificats* pour obtenir le certificat de l' *autorité de certification racine* .

La chaîne de certificats, également appelée *chemin de certification*, est une liste de certificats utilisés pour authentifier une entité. La chaîne, ou chemin, commence par le certificat de cette entité, et chaque certificat de la chaîne est signé par l'entité identifiée par le certificat suivant de la chaîne. La chaîne s'arrête avec un certificat d'autorité de certification racine. Le certificat de l'autorité de certification racine est toujours signé par l'autorité de certification elle-même. Les signatures de tous les certificats de la chaîne doivent être vérifiées jusqu'à ce que le certificat de l'autorité de certification racine soit atteint.

La Figure 4, à la page 14 illustre un chemin de certification entre le propriétaire du certificat et l'autorité de certification racine, où commence la chaîne de confiance.

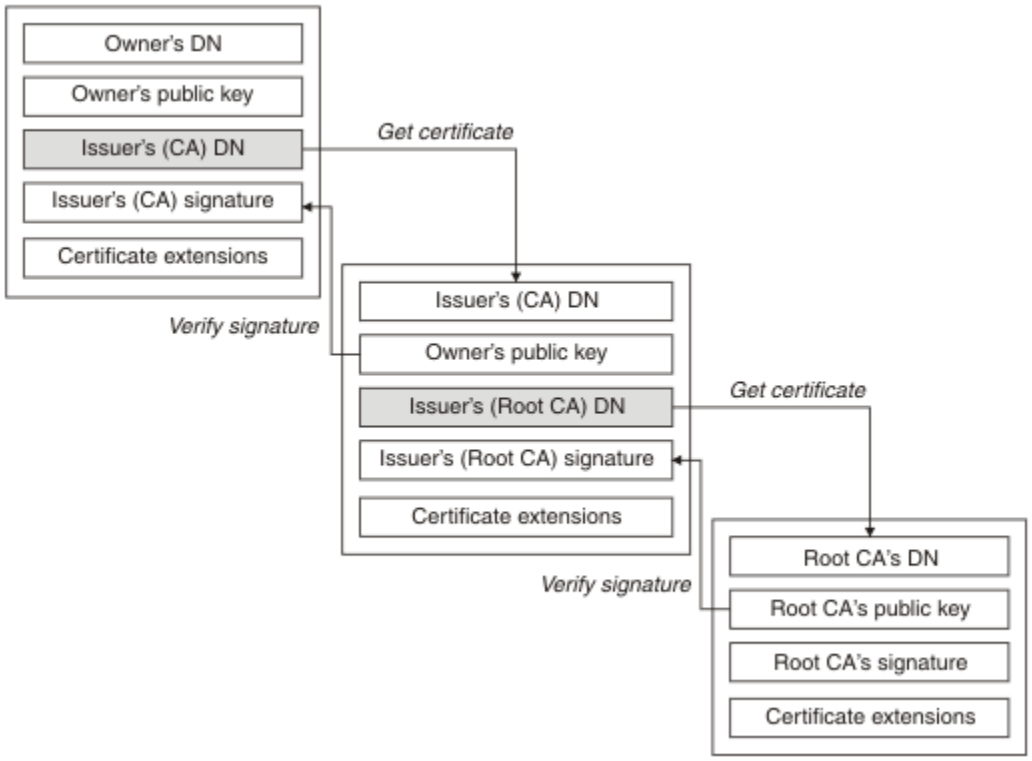


Figure 4. Chaîne de confiance

Chaque certificat peut contenir une ou plusieurs extensions. Un certificat appartenant à une autorité de certification contient généralement une extension BasicConstraints avec l'indicateur isCA défini pour indiquer qu'il est autorisé à signer d'autres certificats.

Lorsque les certificats ne sont plus valides
Les certificats numériques peuvent expirer ou être révoqués.

Les certificats numériques sont délivrés pour une période déterminée et ne sont pas valides après leur date d'expiration.

Les certificats peuvent être révoqués pour diverses raisons, notamment:

- Le propriétaire est parti dans une autre entreprise.
- La clé privée n'est plus secrète.

IBM MQ peut vérifier si un certificat est révoqué en envoyant une demande à un répondeur OCSP (Online Certificate Status Protocol) (sur UNIX, Linux®, and Windows uniquement). Ils peuvent également accéder à une liste de révocation de certificat (CRL) sur un serveur LDAP. Les informations de révocation OCSP et de LRC sont publiées par une autorité de certification. Pour plus d'informations, voir «Utilisation des certificats révoqués», à la page 348.

Public key infrastructure (PKI)

Une infrastructure à clé publique (ICP) est un système d'installations, de politiques et de services qui prend en charge l'utilisation de la cryptographie à clé publique pour authentifier les parties impliquées dans une transaction.

Il n'existe pas de norme unique qui définit les composants d'une infrastructure à clé publique, mais une ICP comprend généralement des autorités de certification (AC) et des autorités d'enregistrement (AR). Les autorités de certification fournissent les services suivants:

- Emission de certificats numériques
- Validation des certificats numériques
- Révocation de certificats numériques

- Distribution de clés publiques

Les normes X.509 constituent la base de l'infrastructure à clé publique conforme aux normes de l'industrie.

Pour plus d'informations sur les certificats numériques et les autorités de certification, voir «Certificats numériques», à la page 9 . Les autorités de certification vérifient les informations fournies lorsque des certificats numériques sont demandés. Si l'autorité de certification vérifie ces informations, l'autorité de certification peut émettre un certificat numérique au demandeur.

Une infrastructure PKI peut également fournir des outils pour la gestion des certificats numériques et des clés publiques. Une infrastructure PKI est parfois décrite comme une *hiérarchie de confiance* pour la gestion des certificats numériques, mais la plupart des définitions incluent des services supplémentaires. Certaines définitions comprennent des services de chiffrement et de signature numérique, mais ces services ne sont pas essentiels au fonctionnement d'une ICP.

Protocole de sécurité cryptographique TLS

Les protocoles cryptographiques fournissent des connexions sécurisées, permettant à deux parties de communiquer avec la confidentialité et l'intégrité des données. Le protocole TLS (Transport Layer Security) a évolué à partir de celui de la couche SSL (Secure Sockets Layer). IBM MQ prend en charge TLS.

Les objectifs principaux des deux protocoles sont de garantir la confidentialité (parfois appelée *confidentialité*), l'intégrité des données, l'identification et l'authentification à l'aide de certificats numériques.

Bien que les deux protocoles soient similaires, les différences sont suffisamment importantes pour que SSL 3.0 et les différentes versions de TLS n'interopèrent pas.

Concepts associés

«Protocoles de sécurité TLS dans IBM MQ», à la page 24
IBM MQ prend en charge le protocole TLS (Transport Layer Security) pour fournir une sécurité de niveau de liaison pour les canaux de message et les canaux MQI.

Concepts TLS (Transport Layer Security)

Le protocole TLS permet à deux parties de s'identifier et de s'authentifier et de communiquer avec la confidentialité et l'intégrité des données. Le protocole TLS a évolué à partir du protocole Netscape SSL 3.0 , mais TLS et SSL n'interopèrent pas.

Le protocole TLS assure la sécurité des communications sur Internet et permet aux applications client/serveur de communiquer de manière confidentielle et fiable. Les protocoles ont deux couches: un protocole d'enregistrement et un protocole d'établissement de liaison, et celles-ci sont superposées au-dessus d'un protocole de transport tel que TCP/IP. Ils utilisent tous deux des techniques de cryptographie asymétrique et symétrique.

Une connexion TLS est initiée par une application qui devient le client TLS. L'application qui reçoit la connexion devient le serveur TLS. Chaque nouvelle session commence par un établissement de liaison, tel que défini par les protocoles TLS.

La liste complète des CipherSpecs pris en charge par IBM MQ est disponible à l'adresse «Activation des CipherSpecs», à la page 434.

Pour plus d'informations sur le protocole SSL, voir les informations fournies à l'adresse https://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt. Pour plus d'informations sur le protocole TLS, voir les informations fournies par le groupe de travail TLS sur le site Web de l'Internet Engineering Task Force à l'adresse https://www.ietf.org

Présentation de l'établissement de liaison SSL/TLS

L'établissement de liaison SSL/TLS permet au client et au serveur TLS d'établir les clés secrètes avec lesquelles ils communiquent.

Cette section fournit un récapitulatif des étapes permettant au client et au serveur TLS de communiquer entre eux.

- Convenir de la version du protocole à utiliser.
- Sélectionnez des algorithmes de cryptographie.
- Authentifiez-vous les uns les autres en échangeant et en validant des certificats numériques.
- Utilisez des techniques de chiffrement asymétrique pour générer une clé secrète partagée, ce qui évite le problème de distribution des clés. TLS utilise ensuite la clé partagée pour le chiffrement symétrique des messages, qui est plus rapide que le chiffrement asymétrique.

Pour plus d'informations sur les algorithmes de cryptographie et les certificats numériques, reportez-vous aux informations connexes.

Dans la présentation, les étapes impliquées dans l'établissement de liaison TLS sont les suivantes:

1. Le client TLS envoie un message "client hello" qui répertorie les informations cryptographiques telles que la version TLS et, dans l'ordre de préférence du client, les CipherSuites prises en charge par le client. Le message contient également une chaîne d'octets aléatoire qui est utilisée dans les calculs ultérieurs. Le protocole permet au "client hello" d'inclure les méthodes de compression de données prises en charge par le client.
2. Le serveur TLS répond avec un message "server hello" qui contient la suite de chiffrement CipherSuite choisie par le serveur dans la liste fournie par le client, l'ID de session et une autre chaîne d'octets aléatoires. Le serveur envoie également son certificat numérique. Si le serveur requiert un certificat numérique pour l'authentification du client, il envoie une "demande de certificat client" qui inclut une liste des types de certificat pris en charge et des noms distinctifs des autorités de certification (CA) acceptables.
3. Le client TLS vérifie le certificat numérique du serveur. Pour plus d'informations, voir «Comment TLS fournit l'identification, l'authentification, la confidentialité et l'intégrité», à la page 17.
4. Le client TLS envoie la chaîne d'octets aléatoires qui permet au client et au serveur de calculer la clé secrète à utiliser pour le chiffrement des données de message suivantes. La chaîne d'octets aléatoires elle-même est chiffrée avec la clé publique du serveur.
5. Si le serveur TLS a envoyé une "demande de certificat client", le client envoie une chaîne d'octets aléatoire chiffrée avec la clé privée du client, ainsi que le certificat numérique du client, ou une "alerte de non-certificat numérique". Cette alerte n'est qu'un avertissement, mais avec certaines implémentations, l'établissement de liaison échoue si l'authentification du client est obligatoire.
6. Le serveur TLS vérifie le certificat du client. Pour plus d'informations, voir «Comment TLS fournit l'identification, l'authentification, la confidentialité et l'intégrité», à la page 17.
7. Le client TLS envoie au serveur un message "finished" , qui est chiffré avec la clé secrète, indiquant que la partie client de l'établissement de liaison est terminée.
8. Le serveur TLS envoie au client un message "finished" , qui est chiffré avec la clé secrète, indiquant que la partie serveur de l'établissement de liaison est terminée.
9. Pendant la durée de la session TLS, le serveur et le client peuvent désormais échanger des messages qui sont chiffrés de manière symétrique avec la clé secrète partagée.

La Figure 5, à la page 17 illustre l'établissement de liaison TLS.

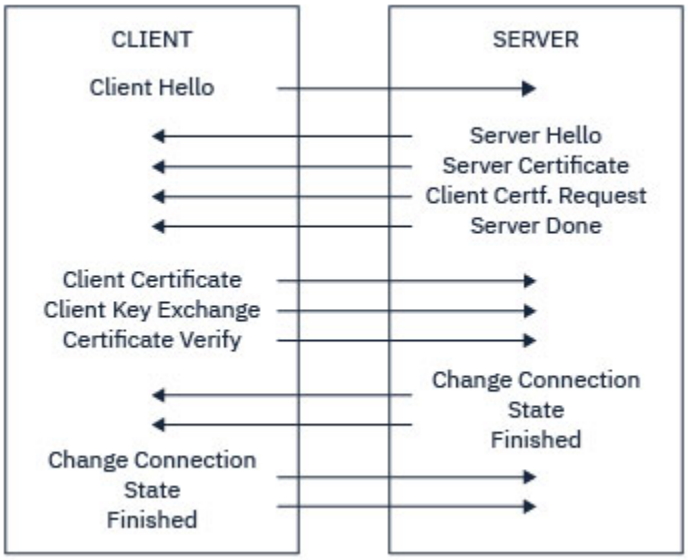


Figure 5. Présentation de l'établissement de liaison TLS

Comment TLS fournit l'identification, l'authentification, la confidentialité et l'intégrité

Lors de l'authentification du client et du serveur, une étape exige que les données soient chiffrées avec l'une des clés d'une paire de clés asymétriques et déchiffrées avec l'autre clé de la paire. Un résumé de message est utilisé pour assurer l'intégrité.

Pour une présentation des étapes impliquées dans l'établissement de liaison TLS, voir «Présentation de l'établissement de liaison SSL/TLS», à la page 15.

Comment TLS fournit l'authentification

Pour l'authentification du serveur, le client utilise la clé publique du serveur pour chiffrer les données utilisées pour calculer la clé secrète. Le serveur ne peut générer la clé secrète que s'il peut déchiffrer ces données avec la clé privée appropriée.

Pour l'authentification client, le serveur utilise la clé publique dans le certificat client pour déchiffrer les données que le client envoie lors de l'étape «5», à la page 16 de l'établissement de liaison. L'échange des messages terminés qui sont chiffrés avec la clé secrète (étapes «7», à la page 16 et «8», à la page 16 de la présentation) confirme que l'authentification est terminée.

Si l'une des étapes d'authentification échoue, l'établissement de liaison échoue et la session se termine.

L'échange de certificats numériques lors de l'établissement de liaison TLS fait partie du processus d'authentification. Pour plus d'informations sur la façon dont les certificats fournissent une protection contre l'usurpation d'identité, reportez-vous aux informations connexes. Les certificats requis sont les suivants, où l'autorité de certification X émet le certificat sur le client TLS et l'autorité de certification Y émet le certificat sur le serveur TLS:

Pour l'authentification de serveur uniquement, le serveur TLS a besoin de:

- Certificat personnel émis sur le serveur par l'autorité de certification Y
- Clé privée du serveur

et le client TLS a besoin:

- Le certificat de l'autorité de certification pour l'autorité de certification Y

Si le serveur TLS requiert une authentification client, le serveur vérifie l'identité du client en vérifiant le certificat numérique du client avec la clé publique de l'autorité de certification qui a émis le certificat personnel au client, en l'occurrence l'autorité de certification X. Pour l'authentification du serveur et du client, le serveur a besoin:

- Certificat personnel émis sur le serveur par l'autorité de certification Y
- Clé privée du serveur
- Le certificat de l'autorité de certification pour l'autorité de certification X

et le client a besoin:

- Certificat personnel émis pour le client par l'autorité de certification X
- Clé privée du client
- Le certificat de l'autorité de certification pour l'autorité de certification Y

Le serveur et le client TLS peuvent avoir besoin d'autres certificats de l'autorité de certification pour former une chaîne de certificats pour le certificat de l'autorité de certification racine. Pour plus d'informations sur les chaînes de certificats, reportez-vous aux informations associées.

Ce qui se passe lors de la vérification de certificat

Comme indiqué dans les étapes «3», à la page 16 et «6», à la page 16 de la présentation, le client TLS vérifie le certificat du serveur et le serveur TLS vérifie le certificat du client. Cette vérification comporte quatre aspects:

1. La signature numérique est vérifiée (voir «Signatures numériques dans SSL/TLS», à la page 19).
2. La chaîne de certificats est vérifiée ; vous devez disposer de certificats d'autorité de certification intermédiaires (voir «Fonctionnement des chaînes de certificats», à la page 13).
3. Les dates d'expiration et d'activation ainsi que la période de validité sont vérifiées.
4. Le statut de révocation du certificat est vérifié (voir «Utilisation des certificats révoqués», à la page 348).

Réinitialisation de la clé secrète

Lors de l'établissement d'une liaison TLS, une *clé secrète* est générée pour chiffrer les données entre le client et le serveur TLS. La clé secrète est utilisée dans une formule mathématique qui est appliquée aux données pour transformer du texte en clair en texte chiffré illisible et du texte chiffré en texte en clair.

La clé secrète est générée à partir du texte aléatoire envoyé dans le cadre de l'établissement de liaison et est utilisée pour chiffrer du texte en clair en texte chiffré. La clé secrète est également utilisée dans l'algorithme MAC (Message Authentication Code), qui est utilisé pour déterminer si un message a été modifié. Pour plus d'informations, voir «Historiques des messages et signatures numériques», à la page 9.

Si la clé secrète est découverte, le texte en clair d'un message peut être déchiffré à partir du texte chiffré, ou le résumé du message peut être calculé, ce qui permet de modifier les messages sans détection. Même pour un algorithme complexe, le texte en clair peut éventuellement être découvert en appliquant chaque transformation mathématique possible au texte chiffré. Pour minimiser la quantité de données pouvant être déchiffrées ou modifiées si la clé secrète est rompue, la clé secrète peut être renégociée périodiquement. Lorsque la clé secrète a été renégociée, la clé secrète précédente ne peut plus être utilisée pour déchiffrer les données chiffrées avec la nouvelle clé secrète.

Comment TLS assure la confidentialité

TLS utilise une combinaison de chiffrement symétrique et asymétrique pour garantir la confidentialité des messages. Lors de l'établissement de liaison TLS, le client et le serveur TLS conviennent d'un algorithme de chiffrement et d'une clé secrète partagée à utiliser pour une seule session. Tous les messages transmis entre le client TLS et le serveur sont chiffrés à l'aide de cet algorithme et de cette clé, ce qui garantit que le message reste privé même s'il est intercepté. Etant donné que TLS utilise le chiffrement asymétrique lors du transport de la clé secrète partagée, il n'y a pas de problème de distribution de clé. Pour plus d'informations sur les techniques de chiffrement, voir «Cryptographie», à la page 7.

Comment TLS assure l'intégrité

TLS assure l'intégrité des données en calculant un résumé de message. Pour plus d'informations, voir «Intégrité des données de messages», à la page 472.

L'utilisation de TLS garantit l'intégrité des données, à condition que le CipherSpec de votre définition de canal utilise un algorithme de hachage comme décrit dans le tableau dans «Activation des CipherSpecs», à la page 434.

En particulier, si l'intégrité des données pose problème, vous devez éviter de choisir un CipherSpec dont l'algorithme de hachage est répertorié comme "Aucun". L'utilisation de MD5 est également fortement déconseillée car elle est désormais très ancienne et n'est plus sécurisée pour des raisons pratiques.

CipherSpecs et CipherSuites

Les protocoles de sécurité cryptographique doivent convenir des algorithmes utilisés par une connexion sécurisée. CipherSpecs et CipherSuites définissent des combinaisons spécifiques d'algorithmes.

Un CipherSpec identifie une combinaison d'algorithme de chiffrement et d'algorithme MAC (Message Authentication Code). Les deux extrémités d'une connexion TLS doivent convenir du même CipherSpec pour pouvoir communiquer.

IBM MQ prend en charge le protocole TLS 1.2 . Toutefois, vous pouvez activer les CipherSpecsobsolètes, si vous devez le faire.

Voir «Activation des CipherSpecs», à la page 434 pour plus d'informations sur:

- CipherSpecs pris en charge par IBM MQ
- Comment activer les CipherSpecs SSL 3.0 et TLS 1.0 CipherSpecs

Important : Lorsque vous utilisez des canaux IBM MQ , vous utilisez un CipherSpec. Lorsque vous utilisez des canaux Java , JMS ou MQTT, vous spécifiez une CipherSuite.

Pour plus d'informations sur les CipherSpecs, voir «Activation des CipherSpecs», à la page 434.

Une CipherSuite est une suite d'algorithmes de cryptographie utilisés par une connexion TLS. Une suite comprend trois algorithmes distincts:

- Algorithme d'échange de clés et d'authentification utilisé lors de l'établissement de liaison
- Algorithme de chiffrement utilisé pour chiffrer les données
- Algorithme MAC (Message Authentication Code) utilisé pour générer le résumé de message

Il existe plusieurs options pour chaque composant de la suite, mais seules certaines combinaisons sont valides lorsqu'elles sont spécifiées pour une connexion TLS. Le nom d'une CipherSuite valide définit la combinaison des algorithmes utilisés. Par exemple, CipherSuite TLS_RSA_WITH_AES_128_CBC_SHA spécifie:

- Algorithme d'authentification et d'échange de clés RSA
- L'algorithme de chiffrement AES, utilisant une clé de 128 bits et le mode CBC (cipher block chaining)
- Code d'authentification de message (MAC) SHA-1

Signatures numériques dans SSL/TLS

Une signature numérique est formée par le chiffrement d'une représentation d'un message. Le chiffrement utilise la clé privée du signataire et, pour des raisons d'efficacité, opère généralement sur un résumé de message plutôt que sur le message lui-même.

Les signatures numériques varient avec les données en cours de signature, contrairement aux signatures manuscrites, qui ne dépendent pas du contenu du document en cours de signature. Si deux messages différents sont signés numériquement par la même entité, les deux signatures diffèrent, mais les deux signatures peuvent être vérifiées avec la même clé publique, c'est-à-dire la clé publique de l'entité qui a signé les messages.

Les étapes du processus de signature numérique sont les suivantes:

1. L'expéditeur calcule un résumé de message, puis il chiffre le résumé à l'aide de la clé privée de l'expéditeur, en formant la signature numérique.
2. L'émetteur transmet la signature numérique avec le message.
3. Le récepteur déchiffre la signature numérique à l'aide de la clé publique de l'expéditeur, en régénérant le résumé du message de l'expéditeur.
4. Le récepteur calcule un résumé de message à partir des données de message reçues et vérifie que les deux résumés sont identiques.

La Figure 6, à la page 20 illustre ce processus.

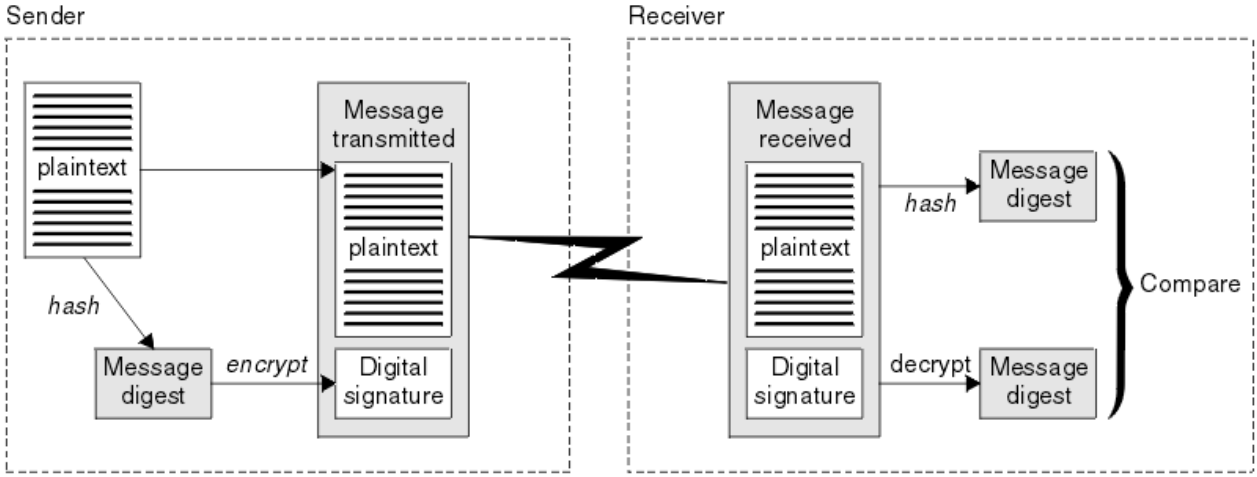


Figure 6. Processus de signature numérique

Si la signature numérique est vérifiée, le récepteur sait que:

- Le message n'a pas été modifié lors de la transmission.
- Le message a été envoyé par l'entité qui prétend l'avoir envoyé.

Les signatures numériques font partie des services d'intégrité et d'authentification. Les signatures numériques fournissent également une preuve de l'origine. Seul l'expéditeur connaît la clé privée, ce qui fournit des preuves solides que l'expéditeur est l'émetteur du message.

Remarque : Vous pouvez également chiffrer le message lui-même, ce qui protège la confidentialité des informations du message.

La norme FIPS (Federal Information Processing Standards)

Le gouvernement américain fournit des conseils techniques sur des systèmes informatiques et sur des systèmes de sécurité, y compris le chiffrement de données. Le National Institute for Standards and Technology (NIST) est un organisme important qui s'occupe des systèmes informatiques et de la sécurité. Il émet des recommandations et établit des normes, notamment la norme FIPS (Federal Information Processing Standards).

L'une de ces normes est la norme FIPS 140-2, qui nécessite l'utilisation d'algorithmes de cryptographie puissants. Elle spécifie également des exigences pour les algorithmes de hachage à utiliser afin de protéger les paquets contre toute modification en transit.

IBM MQ fournit une prise en charge FIPS 140-2 lorsqu'il a été configuré pour le faire.

Avec le temps, les analystes développent des attaques contre les algorithmes de chiffrement et de hachage existants. De nouveaux algorithmes sont adoptés pour résister à ces attaques. La norme FIPS 140-2 est mise à jour régulièrement afin de tenir compte de ces changements.

Concepts associés

«Agence de sécurité nationale (NSA) Suite B Cryptographie», à la page 21
Le gouvernement des États-Unis d'Amérique fournit des conseils techniques sur les systèmes informatiques et la sécurité, y compris le chiffrement des données. La National Security Agency (NSA)

des États-Unis recommande un ensemble d'algorithmes cryptographiques interopérables dans sa norme Suite B.

Agence de sécurité nationale (NSA) Suite B Cryptographie

Le gouvernement des États-Unis d'Amérique fournit des conseils techniques sur les systèmes informatiques et la sécurité, y compris le chiffrement des données. La National Security Agency (NSA) des États-Unis recommande un ensemble d'algorithmes cryptographiques interopérables dans sa norme Suite B.

La norme Suite B spécifie un mode de fonctionnement dans lequel seul un ensemble spécifique d'algorithmes cryptographiques sécurisés est utilisé. La norme Suite B spécifie:

- L'algorithme de chiffrement (AES)
- L'algorithme d'échange de clés (Elliptic Curve Diffie-Hellman, également connu sous le nom d'ECDH)
- L'algorithme de signature numérique (Elliptic Curve Digital Signature Algorithm, également appelé ECDSA)
- Les algorithmes de hachage (SHA-256 ou SHA-384)

De plus, la norme IETF RFC 6460 spécifie des profils compatibles Suite B qui définissent la configuration détaillée de l'application et le comportement nécessaires pour se conformer à la norme Suite B. Il définit deux profils:

1. Profil compatible Suite B à utiliser avec TLS 1.2. Lorsqu'il est configuré pour une opération conforme à la suite B, seul l'ensemble restreint d'algorithmes de cryptographie répertorié est utilisé.
2. Profil de transition à utiliser avec TLS 1.0 ou TLS 1.1. Ce profil permet l'interopérabilité avec les serveurs non conformes à la norme Suite B. Lorsqu'il est configuré pour l'opération de transition Suite B, des algorithmes de chiffrement et de hachage supplémentaires peuvent être utilisés.

La norme Suite B est conceptuellement similaire à la norme FIPS 140-2, car elle restreint l'ensemble des algorithmes de cryptographie activés afin de fournir un niveau de sécurité assuré.

Sous Windows, les systèmes UNIX and Linux , IBM MQ, peuvent être configurés pour être conformes au profil TLS 1.2 compatible Suite B, mais ne prennent pas en charge le profil de transition Suite B. Pour plus d'informations, reportez-vous à la section «NSA Suite B Cryptography dans IBM MQ», à la page 41.

Référence associée

«La norme FIPS (Federal Information Processing Standards)», à la page 20
Le gouvernement américain fournit des conseils techniques sur des systèmes informatiques et sur des systèmes de sécurité, y compris le chiffrement de données. Le National Institute for Standards and Technology (NIST) est un organisme important qui s'occupe des systèmes informatiques et de la sécurité. Il émet des recommandations et établit des normes, notamment la norme FIPS (Federal Information Processing Standards).

IBM MQ mécanismes de sécurité

Cette collection de rubriques explique comment implémenter les différents concepts de sécurité dans IBM MQ.

IBM MQ fournit des mécanismes permettant d'implémenter tous les concepts de sécurité introduits dans «Concepts et mécanismes de sécurité», à la page 5. Ces questions sont abordées plus en détail dans les sections suivantes.

Identification et authentification dans IBM MQ

Dans IBM MQ, vous pouvez implémenter l'identification et l'authentification à l'aide des informations de contexte de message et de l'authentification mutuelle.

Voici quelques exemples d'identification et d'authentification dans un environnement IBM MQ :

- Chaque message peut contenir des informations de *contexte de message* . Ces informations sont conservées dans le descripteur de message. Il peut être généré par le gestionnaire de files d'attente

lorsqu'un message est inséré dans une file d'attente par une application. L'application peut également fournir les informations si l'ID utilisateur associé à l'application est autorisé à le faire.

Les informations de contexte d'un message permettent à l'application de réception de connaître l'émetteur du message. Il contient, par exemple, le nom de l'application qui a inséré le message et l'ID utilisateur associé à l'application.

- Lorsqu'un canal de transmission de messages démarre, il est possible que l'agent MCA (Message Channel Agent) à chaque extrémité du canal authentifie son partenaire. Cette technique est appelée *authentification mutuelle*. Pour l'agent MCA émetteur, il fournit l'assurance que le partenaire auquel il est sur le point d'envoyer des messages est authentique. Pour l'agent MCA récepteur, il existe une assurance similaire qu'il est sur le point de recevoir des messages d'un véritable partenaire.

Concepts associés

«Identification et authentification», à la page 6
L' *identification* est la possibilité d'identifier de manière unique un utilisateur d'un système ou d'une application qui s'exécute dans le système. L' *authentification* est la possibilité de prouver qu'un utilisateur ou une application est réellement qui est cette personne ou ce que cette application prétend être.

Autorisation dans IBM MQ

Vous pouvez utiliser l'autorisation pour limiter les actions que peuvent effectuer des individus ou des applications spécifiques dans votre environnement IBM MQ .

Voici quelques exemples d'autorisation dans un environnement IBM MQ :

- Autoriser uniquement un administrateur autorisé à émettre des commandes pour gérer les ressources IBM MQ .
- Autoriser une application à se connecter à un gestionnaire de files d'attente uniquement si l'ID utilisateur associé à l'application est autorisé à le faire.
- Autoriser une application à ouvrir uniquement les files d'attente nécessaires à sa fonction.
- Autoriser une application à s'abonner uniquement aux rubriques nécessaires à sa fonction.
- Autoriser une application à effectuer uniquement les opérations sur une file d'attente qui sont nécessaires à sa fonction. Par exemple, une application peut n'avoir besoin que de parcourir les messages d'une file d'attente particulière et non d'insérer ou d'extraire des messages.

Pour plus d'informations sur la configuration de l'autorisation, voir «Autorisation de planification», à la page 85 et les sous-rubriques associées.

Concepts associés

«Authorization», à la page 6
L' *autorisation* protège les ressources critiques d'un système en limitant l'accès uniquement aux utilisateurs autorisés et à leurs applications. Il empêche l'utilisation non autorisée d'une ressource ou l'utilisation d'une ressource de manière non autorisée.

Audit dans IBM MQ

IBM MQ peut émettre des messages d'événement pour enregistrer que l'activité inhabituelle a eu lieu.

Voici quelques exemples d'audit dans un environnement IBM MQ :

- Une application tente d'ouvrir une file d'attente qu'elle n'est pas autorisée à ouvrir. Un message d'événement d'instrumentation est émis. En inspectant le message d'événement, vous découvrez que cette tentative a eu lieu et pouvez décider de l'action nécessaire.
- Une application tente d'ouvrir un canal, mais la tentative échoue car SSL n'autorise pas la connexion. Un message d'événement d'instrumentation est émis. En inspectant le message d'événement, vous découvrez que cette tentative a eu lieu et pouvez décider de l'action nécessaire.

Concepts associés

«Audit», à la page 7

L' *audit* est le processus d'enregistrement et de vérification des événements pour détecter si une activité inattendue ou non autorisée a eu lieu ou si une tentative a été effectuée pour effectuer cette activité.

Confidentialité dans IBM MQ

Vous pouvez implémenter la confidentialité dans IBM MQ en chiffrant les messages.

La confidentialité peut être assurée dans un environnement IBM MQ comme suit:

- Une fois qu'un agent MCA émetteur obtient un message d'une file d'attente de transmission, IBM MQ utilise TLS pour chiffrer le message avant qu'il ne soit envoyé sur le réseau à l'agent MCA récepteur. A l'autre extrémité du canal, le message est déchiffré avant que l'agent MCA récepteur ne le place dans sa file d'attente de destination.
- Alors que les messages sont stockés dans une file d'attente locale, les mécanismes de contrôle d'accès fournis par IBM MQ peuvent être considérés comme suffisants pour protéger leur contenu contre toute divulgation non autorisée. Toutefois, pour un niveau de sécurité plus élevé, vous pouvez utiliser Advanced Message Security pour chiffrer les messages stockés dans les files d'attente.
- V 9.1.4 z/OS Les messages stockés dans les files d'attente locales peuvent être chiffrés au repos à l'aide du chiffrement des fichiers z/OS .

  Voir la section Confidentiality for data at rest on IBM MQ for z/OS with data set encryption pour plus d'informations.

Concepts associés

«Confidentialité», à la page 7
Le service de *confidentialité* protège les informations sensibles contre toute divulgation non autorisée.

Intégrité des données dans IBM MQ

Vous pouvez utiliser un service d'intégrité des données pour détecter si un message a été modifié.

L'intégrité des données peut être assurée dans un environnement IBM MQ comme suit:

- Vous pouvez utiliser TLS pour détecter si le contenu d'un message a été délibérément modifié alors qu'il était transmis sur un réseau. Dans TLS, l'algorithme de synthèse de message permet de détecter les messages modifiés en transit.

  Tous les CipherSpecs IBM MQ fournissent un algorithme de synthèse de message, à l'exception de TLS_RSA_WITH_NULL_NULL, qui ne fournit pas l'intégrité des données de message.

  IBM MQ détecte les messages modifiés lors de leur réception ; lors de la réception d'un message modifié, IBM MQ émet un message d'erreur AMQ9661 et le canal s'arrête.
- Lorsque les messages sont stockés dans une file d'attente locale, les mécanismes de contrôle d'accès fournis par IBM MQ peuvent être considérés comme suffisants pour empêcher une modification délibérée du contenu des messages.

  Toutefois, pour un niveau de sécurité plus élevé, vous pouvez utiliser Advanced Message Security pour détecter si le contenu d'un message a été délibérément modifié entre le moment où le message a été inséré dans la file d'attente et le moment où il a été extrait de la file d'attente.

  Lors de la détection d'un message modifié, l'application qui tente de recevoir le message reçoit un code retour 2063 et, si vous utilisez un appel MQGET , le message est déplacé vers SYSTEM.PROTECTION.ERROR.QUEUE

Concepts associés

«Intégrité des données», à la page 7
Le service d' *intégrité des données* détecte s'il y a eu une modification non autorisée des données.

Cryptographie dans IBM MQ

IBM MQ fournit la cryptographie à l'aide du protocole TLS (Transport Security Layer).

Pour plus d'informations, voir «Protocoles de sécurité TLS dans IBM MQ», à la page 24.

Concepts associés

«Concepts cryptographiques», à la page 7
Cette collection de rubriques décrit les concepts de cryptographie applicables à IBM MQ.

Protocoles de sécurité TLS dans IBM MQ

IBM MQ prend en charge le protocole TLS (Transport Layer Security) pour fournir une sécurité de niveau de liaison pour les canaux de message et les canaux MQI.

Les canaux de transmission de messages et les canaux MQI peuvent utiliser le protocole TLS pour assurer la sécurité au niveau de la liaison. Un agent MCA appelant est un client TLS et un agent MCA répondeur est un serveur TLS. IBM MQ prend en charge TLS 1.0 et TLS 1.2. Vous pouvez spécifier les algorithmes de cryptographie utilisés par le protocole TLS en fournissant un CipherSpec dans la définition de canal.

Remarque : Depuis IBM MQ 8.0.0 Fix Pack 2, le protocole SSLv3 et l'utilisation de certains CipherSpecs IBM MQ sont obsolètes. Pour plus d'informations, voir Dépréciation: protocole SSLv3.

Vous pouvez utiliser les paramètres SECPROT et SSLCIPH pour afficher le protocole de sécurité et CipherSpec utilisé sur un canal.

A chaque extrémité d'un canal de transmission de messages et à l'extrémité serveur d'un canal MQI, l'agent MCA agit pour le compte du gestionnaire de files d'attente auquel il est connecté. Lors de l'établissement de liaison TLS, l'agent MCA envoie le certificat numérique du gestionnaire de files d'attente à son agent MCA partenaire à l'autre extrémité du canal. Le code IBM MQ à l'extrémité client d'un canal MQI agit pour le compte de l'utilisateur de l'application client IBM MQ . Lors de l'établissement de liaison TLS, le code IBM MQ envoie le certificat numérique de l'utilisateur à l'agent MCA à l'extrémité serveur du canal MQI.

Les gestionnaires de files d'attente et les utilisateurs de client IBM MQ ne sont pas tenus d'avoir des certificats numériques personnels qui leur sont associés lorsqu'ils agissent en tant que clients TLS, sauf si SSLCAUTH (REQUIRED) est spécifié côté serveur du canal.

Les certificats numériques sont stockés dans un *référentiel de clés*. L'attribut de gestionnaire de files d'attente **SSLKeyRepository** indique l'emplacement du référentiel de clés qui contient le certificat numérique du gestionnaire de files d'attente. Sur un système client IBM MQ , la variable d'environnement MQSSLKEYR indique l'emplacement du référentiel de clés qui contient le certificat numérique de l'utilisateur. Une application client IBM MQ peut également spécifier son emplacement dans la zone **KeyRepository** de la structure d'options de configuration TLS, MQSCO, sur un appel MQCONNX. Consultez les rubriques connexes pour plus d'informations sur les référentiels de clés et pour savoir comment spécifier leur emplacement.

Prise en charge de TLS

IBM MQ prend en charge TLS 1.0 et TLS 1.2 en fonction de la plateforme que vous utilisez. Pour plus d'informations sur le protocole TLS, reportez-vous aux informations des sous-rubriques.

IBM i
La prise en charge de TLS fait partie intégrante du système d'exploitation IBM i .

Clients Java et JMS
Ces clients utilisent la machine virtuelle Java pour fournir la prise en charge TLS.

UNIX, Linux, and Windows
La prise en charge de TLS est installée avec IBM MQ.

z/OS
La prise en charge de TLS fait partie intégrante du système d'exploitation z/OS . La prise en charge de TLS sur z/OS est appelée *System SSL*.

Pour plus d'informations sur les prérequis pour la prise en charge de TLS dans IBM MQ , voir Configuration système requise pour IBM MQ.

Concepts associés

«Protocole de sécurité cryptographique TLS», à la page 15
Les protocoles cryptographiques fournissent des connexions sécurisées, permettant à deux parties de communiquer avec la confidentialité et l'intégrité des données. Le protocole TLS (Transport Layer Security) a évolué à partir de celui de la couche SSL (Secure Sockets Layer). IBM MQ prend en charge TLS.

Référentiel de clés SSL/TLS

Une connexion TLS mutuellement authentifiée requiert un référentiel de clés à chaque extrémité de la connexion. Le référentiel de clés inclut des certificats numériques et des clés privées.

Ces informations utilisent le terme général *référentiel de clés* pour décrire le magasin des certificats numériques et leurs clés privées associées. Le référentiel de clés est référencé par des noms différents sur des plateformes et des environnements différents qui prennent en charge TLS:

- IBM i Sous IBM i: *magasin de certificats*
- Sous Java et JMS: *magasin de clés* et *magasin de clés de confiance*
- ULW Sous UNIX, Linux, and Windows: *fichier de base de données de clés*
- z/OS Sous z/OS: *keyring*

Pour plus d'informations, reportez-vous aux sections «Certificats numériques», à la page 9 et «Concepts TLS (Transport Layer Security)», à la page 15.

Une connexion TLS mutuellement authentifiée requiert un référentiel de clés à chaque extrémité de la connexion. Le référentiel de clés peut contenir les certificats et demandes suivants:

- Un certain nombre de certificats de l'autorité de certification provenant de différentes autorités de certification qui permettent au gestionnaire de files d'attente ou au client de vérifier les certificats qu'il reçoit de son partenaire à l'extrémité éloignée de la connexion. Les certificats individuels peuvent se trouver dans une chaîne de certificats.
- Un ou plusieurs certificats personnels reçus d'une autorité de certification. Vous associez un certificat personnel distinct à chaque gestionnaire de files d'attente ou à IBM MQ MQI client. Les certificats personnels sont essentiels sur un client TLS si une authentification mutuelle est requise. Si l'authentification mutuelle n'est pas requise, les certificats personnels ne sont pas nécessaires sur le client. Le référentiel de clés peut également contenir la clé privée correspondant à chaque certificat personnel.
- Demandes de certificat qui attendent d'être signées par un certificat de l'autorité de certification digne de confiance.

Pour plus d'informations sur la protection de votre référentiel de clés, voir «Protection des référentiels de clés IBM MQ», à la page 26.

L'emplacement du référentiel de clés dépend de la plateforme que vous utilisez:

IBM i

Le référentiel de clés est un magasin de certificats. Le magasin de certificats de système par défaut se trouve à l'emplacement `/QIBM/UserData/ICSS/Cert/Server/Default` dans le système de fichiers intégré (IFS). IBM MQ stocke le mot de passe du magasin de certificats dans un *fichier de mot de passe secret*. Par exemple, le fichier de dissimulation du gestionnaire de files d'attente `QM1` est `/QIBM/UserData/mqm/qmgrs/QM1/ssl/Stash.sth`.

Vous pouvez également indiquer que le magasin de certificats de système IBM i doit être utilisé à la place. Pour ce faire, remplacez la valeur de l'attribut **SSLKEYR** du gestionnaire de files d'attente par *SYSTEM. Cette valeur indique que le gestionnaire de files d'attente doit utiliser le magasin de certificats du système et que le gestionnaire de files d'attente est enregistré pour être utilisé en tant qu'application avec Digital Certificate Manager (DCM).

Le magasin de certificats contient également la clé privée du gestionnaire de files d'attente.

UNIX, Linux, and Windows systèmes

Le référentiel de clés est un fichier de base de données de clés. Le nom du fichier de la base de données de clés doit avoir l'extension `.kdb`. Par exemple, sous UNIX and Linux, le fichier de base de données de clés par défaut pour le gestionnaire de files d'attente `QM1` est `/var/mqm/qmgrs/QM1/ssl/key.kdb`. Si IBM MQ est installé dans l'emplacement par défaut, le chemin équivalent sous Windows est `C:\ProgramData\IBM\MQ\Qmgrs\QM1\ssl\key.kdb`.

Chaque fichier de base de données de clés est associé à un fichier de mot de passe secret. Ce fichier contient les mots de passe codés qui permettent aux programmes d'accéder à la base de données de clés. Le fichier de mot de passe secret doit se trouver dans le même répertoire et avoir le même radical de fichier que la base de données de clés, et doit se terminer par le suffixe `.sth`, par exemple `/var/mqm/qmgrs/QM1/ssl/key.sth`

Remarque : Les cartes matérielles de cryptographie PKCS #11 peuvent contenir les certificats et les clés qui sont conservés dans un fichier de base de données de clés. Lorsque des certificats et des clés sont détenus sur des cartes PKCS #11 , IBM MQ a toujours besoin d'accéder à la fois à un fichier de base de données de clés et à un fichier de mot de passe secret.

Sur les systèmes UNIX et Windows , la base de données de clés contient également la clé privée du certificat personnel associé au gestionnaire de files d'attente ou à IBM MQ MQI client.

z/OS

Les certificats sont stockés dans un fichier de clés dans z/OS.

D'autres gestionnaires de sécurité externes utilisent également des fichiers de clés pour stocker des certificats.

Les clés privées sont gérées par RACF.

Protection des référentiels de clés IBM MQ

Le référentiel de clés pour IBM MQ est un fichier. Assurez-vous que seul l'utilisateur prévu peut accéder au fichier de référentiel de clés. Cela empêche un intrus ou un autre utilisateur non autorisé de copier le fichier de référentiel de clés sur un autre système, puis de configurer un ID utilisateur identique sur ce système pour simuler les droits d'accès de l'utilisateur prévu.

Les droits sur les fichiers dépendent de l'umask de l'utilisateur et de l'outil utilisé. Sous Windows, les comptes IBM MQ requièrent des droits `BypassTraverseChecking` , ce qui signifie que les droits des dossiers dans le chemin d'accès au fichier n'ont aucun effet.

Vérifiez les droits d'accès aux fichiers du référentiel de clés et assurez-vous que les fichiers et le dossier qui les contient ne sont pas lisibles par tout le monde, de préférence pas même par groupe.

La mise en lecture seule du magasin de clés est recommandée, quel que soit le système que vous utilisez, seul l'administrateur étant autorisé à activer les opérations d'écriture afin d'effectuer la maintenance.

Dans la pratique, vous devez protéger tous les magasins de clés, quel que soit l'emplacement et s'ils sont protégés par mot de passe ou non ; protégez les référentiels de clés.

Labels de certificat numérique, compréhension des exigences

Lors de la configuration de TLS pour utiliser des certificats numériques, il peut y avoir des exigences de libellé spécifiques que vous devez respecter, en fonction de la plateforme utilisée et de la méthode que vous utilisez pour vous connecter.

Qu'est-ce que le label de certificat?

Un label de certificat est un identificateur unique représentant un certificat numérique stocké dans un référentiel de clés et fournit un nom lisible par l'utilisateur qui permet de faire référence à un certificat particulier lors de l'exécution de fonctions de gestion de clés. Vous affectez le libellé de certificat lors de l'ajout d'un certificat à un référentiel de clés pour la première fois.

Le libellé de certificat est distinct des zones **`Subject Distinguished Name`** ou **`Subject Common Name`** du certificat. Notez que **`Subject Distinguished Name`** et **`Subject Common Name`** sont des

zones du certificat lui-même. Elles sont définies lors de la création du certificat et ne peuvent pas être modifiées. Toutefois, si nécessaire, vous pouvez modifier le libellé associé à un certificat numérique.

Syntaxe du libellé de certificat

Un label de certificat peut contenir des lettres, des chiffres et des signes de ponctuation avec les conditions suivantes:

- Multi Le libellé de certificat peut contenir jusqu'à 64 caractères.
- z/OS Le libellé de certificat peut contenir jusqu'à 32 caractères.
- Le libellé de certificat peut contenir des espaces.
- Les libellés sont sensibles à la casse.
- Sur les systèmes qui utilisent EBCDIC katakana, vous ne pouvez pas utiliser de caractères minuscules.

Des exigences supplémentaires pour les valeurs de label de certificat sont spécifiées dans les sections suivantes.

Comment le label de certificat est-il utilisé?

IBM MQ utilise des libellés de certificat pour localiser un certificat personnel envoyé lors de l'établissement de liaison TLS. Cela élimine l'ambiguïté lorsque plusieurs certificats personnels existent dans le référentiel de clés.

Vous pouvez définir le libellé de certificat sur une valeur de votre choix. Si vous ne définissez pas de valeur, un libellé par défaut est utilisé, qui suit une convention de dénomination en fonction de la plateforme que vous utilisez. Pour plus de détails, voir les sections suivantes sur des plateformes particulières.

Remarques :

1. Vous ne pouvez pas définir vous-même le libellé de certificat sur les systèmes Java ou JMS .
2. Les canaux définis automatiquement créés par un exit de définition automatique de canal (CHAD) ne peuvent pas définir le libellé de certificat car l'établissement de liaison TLS a eu lieu au moment de la création du canal. La définition du libellé de certificat dans un exit CHAD pour les canaux entrants n'a aucun effet.

Dans ce contexte, un client TLS fait référence au partenaire de connexion qui initie l'établissement de liaison, qui peut être un client IBM MQ ou un autre gestionnaire de files d'attente.

Lors de l'établissement de liaison TLS, le client TLS obtient toujours un certificat numérique du serveur et le valide. Avec l'implémentation IBM MQ , le serveur TLS demande toujours un certificat au client et le client fournit toujours un certificat au serveur s'il en trouve un. Si le client ne parvient pas à localiser un certificat personnel, il envoie une réponse `no certificate` au serveur.

Le serveur TLS valide toujours le certificat client si celui-ci est envoyé. Si le client n'envoie pas de certificat, l'authentification échoue si l'extrémité du canal qui agit en tant que serveur TLS est définie avec le paramètre **SSLCAUTH** défini sur *REQUIRED* ou une valeur de paramètre **SSLPEER** définie.

Notez que les canaux entrants (y compris les canaux récepteur, demandeur, récepteur de cluster, serveur non qualifié et connexion serveur) n'envoient le certificat configuré que si la version IBM MQ de l'homologue distant prend entièrement en charge la configuration des libellés de certificat et que le canal utilise un CipherSpecTLS.

Un canal serveur non qualifié est un canal dont la zone CONNAME n'est pas définie.

Dans tous les autres cas, le paramètre **CERTLABL** du gestionnaire de files d'attente détermine le certificat envoyé. En particulier, les éléments suivants ne reçoivent que le certificat configuré par le paramètre **CERTLABL** du gestionnaire de files d'attente, quelle que soit la valeur de libellé spécifique au canal:

- Avant IBM MQ 9.1.1, tous les clients Java et JMS en cours.

- V 9.1.1 Des clients IBM MQ 9.1.1, Java et JMS prenant en charge SNI (Server Name Indication), c'est-à-dire des certificats canal par canal.
- Versions de IBM MQ antérieures à IBM MQ 8.0.
- Clients .NET gérés

En outre, le certificat utilisé par un canal doit être approprié pour le canal CipherSpec -voir «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45 pour plus d'informations.

IBM MQ 8.0 et les versions ultérieures prennent en charge l'utilisation de plusieurs certificats sur le même gestionnaire de files d'attente, à l'aide d'un libellé de certificat par canal, spécifié à l'aide de l'attribut **CERTLABL** sur la définition de canal. Les canaux entrants dans le gestionnaire de files d'attente (par exemple, la connexion serveur ou le récepteur) dépendent de la détection du nom de canal à l'aide de TLS Server Name Indication (SNI), afin de présenter le certificat correct du gestionnaire de files d'attente.

Si un canal se connecte au gestionnaire de files d'attente de destination via IBM MQ Internet Pass-Thru (MQIPT) et que **SSLServer** et **SSLClient** sont définis pour la route MQIPT , il existe deux sessions TLS distinctes entre les noeuds finaux et les données SNI ne transitent pas par l'interruption de session. Cela empêche l'utilisation d'un certificat par canal sur le gestionnaire de files d'attente de destination pour la connexion TLS entre MQIPT et le gestionnaire de files d'attente. Pour utiliser un certificat par canal sur le gestionnaire de files d'attente de destination, pour une connexion TLS qui passe par MQIPT, la route MQIPT doit utiliser le mode proxy TLS, qui transmet tous les flux de contrôle TLS intacts, y compris le nom SNI. Pour plus d'informations sur le support TLS dans MQIPT, voir Support SSL/TLS.

Les certificats utilisés pour les connexions TLS qui sont arrêtées ou initiées par MQIPT peuvent être configurés individuellement pour chaque route, par exemple à l'aide des propriétés de route **SSLServerSiteLabel** et **SSLClientSiteLabel** .

Pour plus d'informations sur la connexion d'un gestionnaire de files d'attente à l'aide de l'authentification unidirectionnelle, c'est-à-dire lorsque le client TLS n'envoie pas de certificat, voir Connexion de deux gestionnaires de files d'attente à l'aide de l'authentification unidirectionnelle.

Systèmes Multiplatforms

Multi

Sous Multiplateformes, le serveur TLS envoie un certificat au client.

Pour les gestionnaires de files d'attente et les clients respectivement, une valeur non vide est recherchée dans l'ordre dans les sources suivantes. La première valeur non vide détermine le libellé de certificat. Le libellé de certificat doit exister dans le référentiel de clés. Si aucun certificat correspondant dans la casse et le format corrects ne correspond à un libellé, une erreur se produit et l'établissement de liaison TLS échoue.

Gestionnaires de files d'attente

1. Attribut de libellé de certificat de canal **CERTLABL**.
2. Attribut de label de certificat de gestionnaire de files d'attente **CERTLABL**.
3. Une valeur par défaut, au format `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ajouté, toutes en minuscules. Par exemple, pour un gestionnaire de files d'attente nommé `QM1`, le libellé de certificat par défaut est `ibmwebspheremqqm1`.

IBM MQ clients

1. Attribut de libellé de certificat **CERTLABL** dans la définition de canal CLNTCONN.
2. Attribut **CertificateLabel** de la structure MQSCO.
3. Variable d'environnement **MQCERTLABL**.
4. Attribut `.ini` file (dans sa section SSL) **CertificateLabel** du client

5. Valeur par défaut, au format: `ibmwebspheremq` avec l'ID utilisateur que l'application client exécute comme ajouté, le tout en minuscules. Par exemple, pour un ID utilisateur USER1, le libellé de certificat par défaut est `ibmwebspheremquser1`.

z/OS systèmes

IBM MQ Les clients ne sont pas pris en charge sous z/OS. Toutefois, un gestionnaire de files d'attente z/OS peut jouer le rôle de client TLS lors du lancement d'une connexion ou de serveur TLS lors de l'acceptation d'une demande de connexion. Les exigences de label de certificat pour les gestionnaires de files d'attente z/OS s'appliquent dans ces deux rôles et diffèrent des exigences sur Multiplateformes.

Pour les gestionnaires de files d'attente et les clients respectivement, une valeur non vide est recherchée dans l'ordre dans les sources suivantes. La première valeur non vide détermine le libellé de certificat. Le libellé de certificat doit exister dans le référentiel de clés. Si aucun certificat correspondant dans la casse et le format corrects ne correspond à un libellé, une erreur se produit et l'établissement de liaison TLS échoue.

1. Attribut de libellé de certificat de canal, **CERTLABL**.
2. S'il est partagé, l'attribut de label de certificat de groupe de partage de files d'attente, **CERTQSGL**.

   S'il n'est pas partagé, l'attribut de label de certificat du gestionnaire de files d'attente, **CERTLABL**.
3. Valeur par défaut, au format: `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente ajouté. Notez que cette chaîne est sensible à la casse et doit être écrite comme indiqué. Par exemple, pour un gestionnaire de files d'attente nommé `QM1`, le libellé de certificat par défaut est `ibmWebSphereMQQM1`.
4. Si aucun certificat n'est trouvé avec le format de l'option «3», à la page 29, IBM MQ tente d'utiliser le certificat marqué comme certificat par défaut dans le fichier de clés.

Pour plus d'informations sur l'affichage du référentiel de clés, voir «Localisation du référentiel de clés d'un gestionnaire de files d'attente sous z/OS», à la page 329.

Clients IBM MQ Java et IBM MQ JMS

Les clients IBM MQ Java et IBM MQ JMS utilisent les fonctions de leur fournisseur JSSE (Java Secure Socket Extension) pour sélectionner un certificat personnel lors de l'établissement de liaison TLS et ne sont donc pas soumis aux exigences de label de certificat.

Le comportement par défaut est que le client JSSE itère via les certificats du référentiel de clés, en sélectionnant le premier certificat personnel acceptable trouvé. Cependant, ce comportement n'est qu'une valeur par défaut et dépend de l'implémentation du fournisseur JSSE.

En outre, l'interface JSSE est hautement personnalisable via la configuration et l'accès direct lors de l'exécution par l'application. Pour plus de détails, consultez la documentation fournie par votre fournisseur JSSE.

Pour le traitement des incidents ou pour mieux comprendre l'établissement de liaison effectué par l'application client IBM MQ Java en association avec votre fournisseur JSSE spécifique, vous pouvez activer le débogage en définissant `javax.net.debug=ssl` dans l'environnement JVM.

Vous pouvez définir la variable dans l'application, via la configuration ou en entrant `-Djavax.net.debug=ssl` sur la ligne de commande.

Régénération du référentiel de clés du gestionnaire de files d'attente
Lorsque vous modifiez le contenu d'un référentiel de clés, le gestionnaire de files d'attente ne récupère pas immédiatement le nouveau contenu. Pour qu'un gestionnaire de files d'attente puisse utiliser le nouveau contenu du référentiel de clés, vous devez exécuter la commande REFRESH SECURITY TYPE (SSL).

Ce processus est intentionnel et évite que plusieurs canaux en cours d'exécution puissent utiliser des versions différentes d'un référentiel de clés. En tant que contrôle de sécurité, une seule version d'un référentiel de clés peut être chargée par le gestionnaire de files d'attente à la fois.

Pour plus d'informations sur la commande REFRESH SECURITY TYPE (SSL), voir REFRESH SECURITY.

Vous pouvez également actualiser un référentiel de clés à l'aide des commandes PCF ou du IBM MQ Explorer. Pour plus d'informations, voir la commande MQCMD_REFRESH_SECURITY et la rubrique *Actualisation de la sécurité TLS* dans la section IBM MQ Explorer de la documentation de ce produit.

Concepts associés

«Actualisation de la vue d'un client du contenu du référentiel de clés SSL/TLS et des paramètres SSl/TLS», à la page 30
Pour mettre à jour l'application client avec le contenu actualisé du référentiel de clés, vous devez arrêter et redémarrer l'application client.

Actualisation de la vue d'un client du contenu du référentiel de clés SSL/TLS et des paramètres SSl/TLS
Pour mettre à jour l'application client avec le contenu actualisé du référentiel de clés, vous devez arrêter et redémarrer l'application client.

Vous ne pouvez pas actualiser la sécurité sur un client IBM MQ ; il n'existe pas d'équivalent de la commande REFRESH SECURITY TYPE (SSL) pour les clients (voir REFRESH SECURITY) pour plus d'informations.

Vous devez arrêter et redémarrer l'application, chaque fois que vous modifiez le certificat de sécurité, pour mettre à jour l'application client avec le contenu actualisé du référentiel de clés.

Si le redémarrage du canal actualise les configurations et que votre application possède une logique de reconnexion, vous pouvez actualiser la sécurité sur le client en exécutant la commande STOP CHL STATUS (INACTIVE).

Concepts associés

«Régénération du référentiel de clés du gestionnaire de files d'attente», à la page 29
Lorsque vous modifiez le contenu d'un référentiel de clés, le gestionnaire de files d'attente ne récupère pas immédiatement le nouveau contenu. Pour qu'un gestionnaire de files d'attente puisse utiliser le nouveau contenu du référentiel de clés, vous devez exécuter la commande REFRESH SECURITY TYPE (SSL).

Protection par mot de passe MQCSP

Depuis IBM MQ 8.0, vous pouvez envoyer des mots de passe inclus dans la structure MQCSP qui sont protégés, à l'aide de la fonctionnalité IBM MQ , ou chiffrés, à l'aide du chiffrement TLS.

Important : La protection par mot de passe MQCSP est utile à des fins de test et de développement car l'utilisation de la protection par mot de passe MQCSP est plus simple que la configuration du chiffrement TLS, mais pas sécurisée. A des fins de production, vous devez utiliser le chiffrement TLS de préférence à la protection par mot de passe IBM MQ , en particulier lorsque le réseau entre le client et le gestionnaire de files d'attente n'est pas sécurisé, car le chiffrement TLS est plus sécurisé.

Si vous vous souciez précisément du chiffrement utilisé et de la protection qu'il offre, vous devez utiliser le chiffrement TLS complet. Dans ce cas, les algorithmes sont connus du public et vous pouvez sélectionner celui qui convient à votre entreprise à l'aide de l'attribut de canal **SSLCIPH** .

Pour plus d'informations sur la structure MQCSP, voir Structure MQCSP.

La protection par mot de passe est utilisée lorsque toutes les conditions suivantes sont remplies:

- Les deux extrémités de la connexion utilisent IBM MQ 8.0ou une version ultérieure.
- Le canal n'utilise pas le chiffrement TLS. Un canal n'utilise pas le chiffrement TLS si le canal comporte un attribut **SSLCIPH** vide ou si l'attribut **SSLCIPH** est défini sur un CipherSpec qui ne fournit pas de chiffrement. Les chiffrements null, par exemple, NULL_SHA, ne fournissent pas de chiffrement.
- Vous définissez **MQCSP**.**AuthenticationType** vers MQCSP_AUTH_USER_ID_AND_PWD. La définition de cette valeur permet d'évaluer davantage de vérifications pour déterminer si la protection par mot de passe est effectuée. Valeur par défaut de **MQCSP**.**AuthenticationType** est MQCSP_AUTH_NONE.

Avec le paramètre par défaut, aucune protection par mot de passe n'est effectuée. Pour plus d'informations, voir **AuthenticationType**.

- Si le client est IBM MQ Explorer et que le mode de compatibilité d'identification utilisateur n'est pas activé, ce qui n'est pas le cas par défaut. Cette condition s'applique uniquement à IBM MQ Explorer.

Si ces conditions ne sont pas remplies, le mot de passe est envoyé en texte en clair, sauf s'il est interdit par le paramètre de configuration **PasswordProtection** .

Paramètre de configuration PasswordProtection

L'attribut **PasswordProtection** de la section Canaux des fichiers de configuration .ini du client et du gestionnaire de files d'attente peut empêcher l'envoi de mots de passe en texte en clair. L'attribut peut prendre l'une des valeurs suivantes. La valeur par défaut est `compatible`:

compatible
Le mot de passe peut être envoyé en texte en clair si le gestionnaire de files d'attente ou le client exécute une version de IBM MQ antérieure à IBM MQ 8.0. En d'autres mots, les mots de passe en texte clair sont autorisés à des fins de compatibilité.

Donc :

- Le mot de passe est envoyé chiffré par le CipherSpec TLS si le chiffrement TLS est utilisé et que le CipherSpec n'est pas null.
- Le mot de passe est envoyé en texte clair si le gestionnaire de files d'attente ou le client exécute une version de IBM MQ antérieure à IBM MQ 8.0et que le chiffrement TLS n'est pas utilisé. Le mot de passe est envoyé en texte en clair car les versions de IBM MQ antérieures à IBM MQ 8.0 ne peuvent envoyer des mots de passe qu'en texte en clair.
- Le mot de passe est envoyé protégé si le gestionnaire de files d'attente et le client exécutent une version de IBM MQ à l'adresse IBM MQ 8.0 ou une version ultérieure, et si une valeur null CipherSpec est utilisée, ou si le chiffrement TLS n'est pas utilisé. **MQCSP**:**AuthenticationType** doit être défini sur MQCSP_AUTH_USER_ID_AND_PWD.
- La connexion échoue avant l'envoi du mot de passe si le gestionnaire de files d'attente et le client exécutent une version de IBM MQ à la IBM MQ 8.0 ou une version ultérieure, et **MQCSP**.**AuthenticationType** n'est pas défini sur MQCSP_AUTH_USER_ID_AND_PWD.

toujours
Le mot de passe doit être chiffré avec un CipherSpec qui n'est pas un CipherSpecnull ou **MQCSP**.**AuthenticationType** doit être défini sur MQCSP_AUTH_USER_ID_AND_PWD. Sinon, la connexion échoue. En d'autres mots, les mots de passe en texte en clair ne sont pas autorisés.

Donc :

- Le mot de passe est envoyé chiffré par le CipherSpec TLS si le chiffrement TLS est utilisé et que le CipherSpec n'est pas null.
- Le mot de passe est envoyé protégé si le gestionnaire de files d'attente et le client exécutent une version de IBM MQ à la IBM MQ 8.0 ou une version ultérieure et que le chiffrement TLS n'est pas utilisé ou qu'un CipherSpec null est utilisé. **MQCSP**:**AuthenticationType** doit être défini sur MQCSP_AUTH_USER_ID_AND_PWD.
- La connexion échoue avant l'envoi du mot de passe si le gestionnaire de files d'attente ou le client exécute une version de IBM MQ antérieure à IBM MQ 8.0et que le chiffrement TLS n'est pas utilisé. Etant donné que les versions de IBM MQ antérieures à IBM MQ 8.0 peuvent envoyer des mots de passe uniquement en texte en clair et que `always` requiert que le mot de passe soit chiffré ou protégé, la connexion échoue.

facultatif
Le mot de passe peut éventuellement être envoyé protégé, mais il est envoyé en texte en clair si **MQCSP**.**AuthenticationType** n'est pas défini sur MQCSP_AUTH_USER_ID_AND_PWD. En d'autres mots, les mots de passe en texte en clair peuvent être envoyés par n'importe quel client.

Donc :

- Le mot de passe est envoyé chiffré par le CipherSpec TLS si le chiffrement TLS est utilisé et que le CipherSpec n'est pas null.
- Le mot de passe est envoyé en texte en clair si un CipherSpec null est utilisé et **MQCSP**.**AuthenticationType** n'est pas défini sur MQCSP_AUTH_USER_ID_AND_PWD.
- Le mot de passe est envoyé en texte clair si le gestionnaire de files d'attente ou le client exécute une version de IBM MQ antérieure à IBM MQ 8.0et que le chiffrement TLS n'est pas utilisé. Le mot de passe est envoyé en texte en clair car les versions de IBM MQ antérieures à IBM MQ 8.0 ne peuvent envoyer des mots de passe qu'en texte en clair.
- Le mot de passe est envoyé protégé si le gestionnaire de files d'attente et le client exécutent une version de IBM MQ sur IBM MQ 8.0 ou une version ultérieure, si le chiffrement TLS n'est pas utilisé ou si un CipherSpec null est utilisé et **MQCSP**.**AuthenticationType** est défini sur MQCSP_AUTH_USER_ID_AND_PWD.

avertissement
Les mots de passe en texte en clair peuvent être envoyés par n'importe quel client. Si un mot de passe en texte clair est reçu, un message d'avertissement (AMQ9297) est consigné dans les journaux des erreurs du gestionnaire de files d'attente.

Pour les clients Java et JMS, le comportement de l'attribut **PasswordProtection** varie en fonction du choix d'utiliser le mode de compatibilité ou le mode MQCSP:

- Si les clients Java et JMS fonctionnent en mode compatibilité, une structure MQCSP n'est pas mise en flux lors du traitement de la connexion. Par conséquent, le comportement de l'attribut **PasswordProtection** est le même que celui décrit pour les clients qui exécutent une version de IBM MQ antérieure à IBM MQ 8.0.
- Si les clients Java et JMS fonctionnent en mode MQCSP, le comportement de l'attribut **PasswordProtection** est le comportement décrit.

Pour plus d'informations sur l'authentification de connexion avec les clients Java et JMS, voir «Authentification de connexion avec le client Java», à la page 79.

Gestionnaire de certificats numériques (DCM)

Utilisez DCM pour gérer les certificats numériques et les clés privées sur IBM i.

Digital Certificate Manager (DCM) vous permet de gérer les certificats numériques et de les utiliser dans des applications sécurisées sur le serveur IBM i . Avec Digital Certificate Manager, vous pouvez demander et traiter des certificats numériques auprès d'autorités de certification ou d'autres tiers. Vous pouvez également agir en tant qu'autorité de certification locale pour créer et gérer des certificats numériques pour vos utilisateurs.

DCM prend également en charge l'utilisation de listes de révocation de certificat (CRL) pour fournir un processus de validation de certificat et d'application plus fort. Vous pouvez utiliser DCM pour définir l'emplacement où réside une CRL d'autorité de certification spécifique sur un serveur LDAP afin que IBM MQ puisse vérifier qu'un certificat spécifique n'a pas été révoqué.

DCM prend en charge et peut détecter automatiquement les certificats dans divers formats. Lorsque DCM détecte un certificat codé PKCS #12 ou un certificat PKCS #7 contenant des données chiffrées, il invite automatiquement l'utilisateur à entrer le mot de passe utilisé pour chiffrer le certificat. DCM n'invite pas les certificats PKCS #7 qui ne contiennent pas de données chiffrées.

DCM fournit une interface utilisateur basée sur un navigateur que vous pouvez utiliser pour gérer les certificats numériques de vos applications et de vos utilisateurs. L'interface utilisateur est divisée en deux cadres principaux: un cadre de navigation et un cadre de tâche.

Vous utilisez le cadre de navigation pour sélectionner les tâches de gestion des certificats ou les applications qui les utilisent. Certaines tâches individuelles sont affichées directement dans le cadre de navigation principal, mais la plupart des tâches du cadre de navigation sont organisées en catégories. Par exemple, Gérer les certificats est une catégorie de tâche qui contient diverses tâches guidées individuelles, telles que Afficher le certificat, Renouveler le certificat et Importer le certificat. Si un élément du cadre de navigation est une catégorie qui contient plusieurs tâches, une flèche s'affiche à

sa gauche. La flèche indique que lorsque vous sélectionnez le lien de catégorie, une liste développée de tâches s'affiche, vous permettant de choisir la tâche à effectuer.

Pour obtenir des informations importantes sur DCM, consultez les publications IBM Redbooks suivantes:

- *IBM i Wired Network Security: OS/400 V5R1 DCM and Cryptographic Enhancements*, SG24-6168. Plus précisément, consultez les annexes pour obtenir des informations essentielles sur la configuration de votre système IBM i en tant qu'autorité de certification locale.
- *AS/400 Internet Security: Developing a Digital Certificate Infrastructure*, SG24-5659. Voir plus particulièrement le chapitre 5. *Digital Certificate Manager for AS/400* , qui décrit AS/400 DCM.

FIPS (Federal Information Processing Standards)

Cette rubrique présente le programme FIPS (Federal Information Processing Standards) Cryptomodule Validation Program du US National Institute of Standards and Technology et les fonctions cryptographiques qui peuvent être utilisées sur les canaux TLS.

Ces informations s'appliquent aux plateformes suivantes:

- ULW UNIX, Linux, and Windows
- z/OS z/OS

ULW Pour plus d'informations sur la conformité FIPS 140-2 d'une connexion TLS IBM MQ sur UNIX, Linux, and Windows, voir «FIPS (Federal Information Processing Standards) pour UNIX, Linux, and Windows», à la page 34.

z/OS Pour plus d'informations sur la conformité FIPS 140-2 d'une connexion TLS IBM MQ sur z/OS, voir «FIPS (Federal Information Processing Standards) pour z/OS», à la page 36.

Si du matériel de cryptographie est présent, les modules de cryptographie utilisés par IBM MQ peuvent être configurés pour être ceux fournis par le fabricant du matériel. Dans ce cas, la configuration est uniquement conforme à la norme FIPS si ces modules cryptographiques sont certifiés FIPS.

Au fil du temps, les normes fédérales de traitement de l'information sont mises à jour pour refléter les nouvelles attaques contre les algorithmes et les protocoles de chiffrement. Par exemple, certains CipherSpecs peuvent ne plus être certifiés FIPS. Lorsque de telles modifications se produisent, IBM MQ est également mis à jour pour implémenter la dernière norme. Vous pouvez alors constater des changements de comportement une fois la maintenance appliquée.

Concepts associés

«Comment indiquer que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur MQI Client», à la page 278
Créez vos référentiels de clés à l'aide d'un logiciel compatible FIPS, puis indiquez que le canal doit utiliser des CipherSpecscertifiés FIPS.

«Utilisation de runmqckm, runmqakmet strmqikm pour gérer les certificats numériques», à la page 294
Sur les systèmes UNIX, Linux, and Windows , gérez les clés et les certificats numériques avec **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide de **runmqckm** (iKeycmd) ou de **runmqakm** (GSKCapiCmd).

Tâches associées

Activation de TLS dans IBM MQ classes for Java

Utilisation du protocole TLS (Transport Layer Security) avec IBM MQ classes for JMS

Référence associée

Propriétés TLS des objets JMS

«La norme FIPS (Federal Information Processing Standards)», à la page 20
Le gouvernement américain fournit des conseils techniques sur des systèmes informatiques et sur des systèmes de sécurité, y compris le chiffrement de données. Le National Institute for Standards and Technology (NIST) est un organisme important qui s'occupe des systèmes informatiques et de la sécurité. Il émet des recommandations et établit des normes, notamment la norme FIPS (Federal Information Processing Standards).

ULW *FIPS (Federal Information Processing Standards) pour UNIX, Linux, and Windows*

Lorsque la cryptographie est requise sur un canal SSL/TLS sur des systèmes Windows, UNIX and Linux , IBM MQ utilise un package de cryptographie appelé IBM Crypto for C (ICC). Sur les plateformes Windows, UNIX and Linux , le logiciel ICC a transmis le programme FIPS (Federal Information Processing Standards) Cryptomodule Validation Program du US National Institute of Standards and Technology, au niveau 140-2.

La conformité FIPS 140-2 d'une connexion TLS IBM MQ sur les systèmes Windows, UNIX and Linux est la suivante:

- Pour tous les canaux de transmission de messages IBM MQ (à l'exception des types de canal CLNTCONN), la connexion est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - La version GSKit ICC installée a été certifiée conforme à la norme FIPS 140-2 sur la version du système d'exploitation et l'architecture matérielle installées.
  - L'attribut SSLFIPS du gestionnaire de files d'attente a été défini sur YES.
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .
- Pour toutes les applications IBM MQ MQI client , la connexion utilise GSKit et est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - La version GSKit ICC installée a été certifiée conforme à la norme FIPS 140-2 sur la version du système d'exploitation et l'architecture matérielle installées.
  - Vous avez indiqué que seule la cryptographie certifiée FIPS doit être utilisée, comme décrit dans la rubrique connexe pour le client MQI.
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .
- Pour les applications IBM MQ classes for Java utilisant le mode client, la connexion utilise les implémentations TLS de l'environnement d'exécution Java et est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - L'environnement d'exécution Java utilisé pour exécuter l'application est conforme à la norme FIPS sur la version de système d'exploitation installée et l'architecture matérielle.
  - Vous avez spécifié que seule la cryptographie certifiée FIPS doit être utilisée, comme décrit dans la rubrique connexe du client Java .
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .
- Pour les applications IBM MQ classes for JMS utilisant le mode client, la connexion utilise les implémentations TLS de l'environnement d'exécution Java et est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - L'environnement d'exécution Java utilisé pour exécuter l'application est conforme à la norme FIPS sur la version de système d'exploitation installée et l'architecture matérielle.
  - Vous avez spécifié que seule la cryptographie certifiée FIPS doit être utilisée, comme décrit dans la rubrique connexe du client JMS .
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .
- Pour les applications client .NET non gérées, la connexion utilise GSKit et est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - La version GSKit ICC installée a été certifiée conforme à la norme FIPS 140-2 sur la version du système d'exploitation et l'architecture matérielle installées.
  - Vous avez spécifié que seule la cryptographie certifiée FIPS doit être utilisée, comme décrit dans la rubrique connexe du client .NET .
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .

- Pour les applications client XMS .NET non gérées, la connexion utilise GSKit et est conforme à la norme FIPS si les conditions suivantes sont remplies:
  - La version GSKit ICC installée a été certifiée conforme à la norme FIPS 140-2 sur la version du système d'exploitation et l'architecture matérielle installées.
  - Vous avez indiqué que seule la cryptographie certifiée FIPS doit être utilisée, comme décrit dans la documentation XMS .NET .
  - Tous les référentiels de clés ont été créés et manipulés uniquement à l'aide de logiciels conformes à la norme FIPS, tels que **runmqakm** avec l'option `-fips` .

Toutes les plateformes prises en charge sont certifiées FIPS 140-2 sauf comme indiqué dans le fichier Readme inclus avec chaque groupe de correctifs ou groupe de mises à jour.

Pour les connexions TLS utilisant GSKit, le composant certifié FIPS 140-2 est nommé *ICC*. Il s'agit de la version de ce composant qui détermine la conformité à la norme GSKit FIPS sur une plateforme donnée. Pour déterminer la version d'ICC actuellement installée, exécutez la commande **dspmqver -p 64 -v** .

Voici un exemple d'extrait de la sortie **dspmqver -p 64 -v** relative à ICC:

```
icc
============
@ (#)CompanyName: IBM Corporation
@(#)LegalTrademarks: IBM
@ (#)FileDescription: IBM Crypto for C-language
@(#)FileVersion:   8.0.0.0
@ (#)LegalCopyright: Eléments sous licence-Propriété d' IBM
@ (#) ICC
@ (#) (C) Copyright IBM Corp. 2002, 2024.
@ (#) Tous droits réservés. Utilisateurs du gouvernement américain
@ (#) Droits restreints-Utilisation, duplication ou divulgation
@ (#) restreint par GSA ADP Schedule Contract avec IBM Corp.
@ (#)ProductName: icc_8.0 (générationGoldCoast ) 100415
@(#)ProductVersion:  8.0.0.0
@(#)ProductInfo:   10/04/15.03:32:19.10/04/15.18:41:51
@ (#) CMVCInfo:
```

L'instruction de certification NIST pour GSKit ICC 8 (incluse dans GSKit 8) est disponible à l'adresse suivante: Cryptographic Module Validation Program.

Si du matériel de cryptographie est présent, les modules de cryptographie utilisés par IBM MQ peuvent être configurés pour être ceux fournis par le fabricant du matériel. Dans ce cas, la configuration est uniquement conforme à la norme FIPS si ces modules cryptographiques sont certifiés FIPS.

Remarque : Les clients SSL et TLS Solaris x86 32 bits configurés pour une opération compatible avec FIPS 140-2 échouent lors de l'exécution sur des systèmes Intel. Cet échec survient car le chargement du fichier de bibliothèque GSKit-Crypto Solaris x86 32 bits compatible avec la norme FIPS 140-2 échoue sur le jeu de circuits Intel. Sur les systèmes affectés, l'erreur AMQ9655 est signalée dans le journal des erreurs du client. Pour résoudre ce problème, désactivez la conformité à la norme FIPS 140-2 ou recompilez l'application client 64 bits, car le code 64 bits n'est pas affecté.

Restrictions DES triple imposées lors d'une opération conforme à la norme FIPS 140-2

Lorsque IBM MQ est configuré pour fonctionner conformément à la norme FIPS 140-2, des restrictions supplémentaires sont appliquées en ce qui concerne Triple DES (3DES) CipherSpecs. Ces restrictions permettent la conformité à la recommandation US NIST SP800-67 .

1. Toutes les parties de la clé Triple DES doivent être uniques.
2. Aucune partie de la clé Triple DES ne peut être une clé Weak, Semi-Weak ou Possiblement-Weak selon les définitions de la norme NIST SP800-67.
3. Vous ne pouvez pas transmettre plus de 32 Go de données via la connexion avant qu'une réinitialisation de clé secrète ne soit nécessaire. Par défaut, IBM MQ ne réinitialise pas la clé de session secrète. Cette réinitialisation doit donc être configurée. L'échec de l'activation de la réinitialisation de la clé secrète lors de l'utilisation d'un CipherSpec Triple DES et de la conformité à la norme FIPS 140-2 entraîne la fermeture de la connexion avec l'erreur AMQ9288 après le

dépassement du nombre maximal d'octets. Pour plus d'informations sur la configuration de la réinitialisation des clés secrètes, voir «Réinitialisation des clés secrètes SSL et TLS», à la page 460.

IBM MQ génère des clés de session Triple DES qui sont déjà conformes aux règles 1 et 2. Toutefois, pour satisfaire à la troisième restriction, vous devez activer la réinitialisation de la clé secrète lors de l'utilisation de CipherSpecs Triple DES dans une configuration FIPS 140-2. Vous pouvez également éviter d'utiliser Triple DES.

Concepts associés

«Comment indiquer que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur MQI Client», à la page 278
Créez vos référentiels de clés à l'aide d'un logiciel compatible FIPS, puis indiquez que le canal doit utiliser des CipherSpecscertifiés FIPS.

«Utilisation de runmqckm, runmqakmet strmqikm pour gérer les certificats numériques», à la page 294
Sur les systèmes UNIX, Linux, and Windows , gérez les clés et les certificats numériques avec **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide de **runmqckm** (iKeycmd) ou de **runmqakm** (GSKCapiCmd).

Tâches associées

Activation de TLS dans IBM MQ classes for Java

Utilisation du protocole TLS (Transport Layer Security) avec IBM MQ classes for JMS

Référence associée

Propriétés TLS des objets JMS

«La norme FIPS (Federal Information Processing Standards)», à la page 20
Le gouvernement américain fournit des conseils techniques sur des systèmes informatiques et sur des systèmes de sécurité, y compris le chiffrement de données. Le National Institute for Standards and Technology (NIST) est un organisme important qui s'occupe des systèmes informatiques et de la sécurité. Il émet des recommandations et établit des normes, notamment la norme FIPS (Federal Information Processing Standards).

z/OS *FIPS (Federal Information Processing Standards) pour z/OS*

Lorsque la cryptographie est requise sur un canal SSL/TLS sous z/OS , IBM MQ utilise un service appelé System SSL. L'objectif de System SSL est de fournir la capacité d'exécution sécurisée dans un mode conçu pour respecter le programme de validation cryptographique FIPS (Federal Information Processing Standards) de l'Institut national des normes et de la technologie des États-Unis, au niveau 140-2.

Lors de l'implémentation de connexions conformes à la norme FIPS 140-2 avec des connexions TLS IBM MQ , il existe un certain nombre de points à prendre en compte:

- Pour activer les canaux de transmission de messages IBM MQ pour la conformité FIPS, vérifiez que les conditions suivantes sont remplies:
  - Le FMID de niveau de sécurité SSL système 3 est installé et configuré (voir Planification de l'installation de IBM MQ).
  - Les modules System SSL sont validés.
  - L'attribut SSLFIPS du gestionnaire de files d'attente a été défini sur **YES**.

Lors de l'exécution en mode FIPS, System SSL utilise CP Assist for Cryptographic Function (CPACF) lorsqu'il est disponible. Les fonctions cryptographiques exécutées par le matériel pris en charge par ICSF lors de l'exécution en mode non FIPS continuent d'être exploitées lors de l'exécution en mode FIPS, à l'exception de la génération de signature RSA qui doit être effectuée dans les logiciels.

Tableau 2. Différences entre le mode FIPS et la prise en charge des algorithmes de mode non FIPS.

	Non FIPS		**Norme FIPS**	
Algorithme	**Tailles de clé**	**Matériel**	**Tailles de clé**	**Matériel**
RC2	40 et 128			

Tableau 2. Différences entre le mode FIPS et la prise en charge des algorithmes de mode non FIPS. (suite)				
	Non FIPS		**Norme FIPS**	
Algorithme	**Tailles de clé**	**Matériel**	**Tailles de clé**	**Matériel**
RC4	40 et 128			
DES	56	x		
TDES	168	x	168	x
AES	128 et 256	x	128 et 256	x
MD5	48			
SHA-1	160	x	160	x
SHA-2	224, 256, 384 et 512	x	224, 256, 384 et 512	x
RSA	512-4096	x	1024-4096	x
DSA	512-1024		1024	
DH	512-2048		2048	

En mode FIPS, System SSL ne peut utiliser que les certificats qui utilisent les algorithmes et les tailles de clé indiqués dans le tableau 1. Lors de la validation du certificat X.509 si un algorithme incompatible avec le mode FIPS est détecté, le certificat ne peut pas être utilisé et est traité comme non valide.

Pour les applications de classes IBM MQ utilisant le mode client dans WebSphere Application Server , voir Prise en charge de la norme FIPS (Federal Information Processing Standard).

Pour plus d'informations sur la configuration du module System SSL, voir Configuration de la vérification du module System SSL .

Référence associée

«La norme FIPS (Federal Information Processing Standards)», à la page 20
Le gouvernement américain fournit des conseils techniques sur des systèmes informatiques et sur des systèmes de sécurité, y compris le chiffrement de données. Le National Institute for Standards and Technology (NIST) est un organisme important qui s'occupe des systèmes informatiques et de la sécurité. Il émet des recommandations et établit des normes, notamment la norme FIPS (Federal Information Processing Standards).

Multi

Vérification de la configuration TLS de votre gestionnaire de files d'attente avec `mqcertck`

La commande **MQCERTCK** est un outil qui permet de rechercher les erreurs courantes dans la configuration TLS de votre gestionnaire de files d'attente et fournit des suggestions pour la résolution des problèmes.

Introduction

La commande **`mqcertck`** vérifie:

- Existence et droits du référentiel de clés du gestionnaire de files d'attente, référencé dans l'attribut **SSLKEYR** du gestionnaire de files d'attente.
- Existence et validité du certificat du gestionnaire de files d'attente, référencé dans l'attribut **CERTLABL** du gestionnaire de files d'attente.
- Existence et validité des certificats référencés dans les attributs **CERTLABL** du canal TLS activé.
- Le référentiel de clés et les certificats des applications client, y compris la vérification des certificats, sont autorisés avec le gestionnaire de files d'attente.

Remarque : La commande **mqcertck** n'est pas disponible sous z/OS ou IBM i.

Utilisation

Pour utiliser la commande **mqcertck** , exécutez la commande mqcertck, ainsi que ses paramètres obligatoires et tous les paramètres facultatifs dont vous avez besoin, à partir d'une ligne de commande.

Voir mqcertck pour une description de la commande et des paramètres utilisés par la commande.

Exemple

Vous venez de terminer la configuration de votre gestionnaire de files d'attente QM1 pour autoriser les connexions TLS des clients se connectant au canal SVRCONN de votre gestionnaire de files d'attente.

Vous utilisez la fonction de certificats multiples et, par conséquent, votre gestionnaire de files d'attente et votre canal ont un libellé de certificat spécifié dans leurs attributs **CERTLABL** . Lors de la création du canal, vous avez fait une erreur dans l'attribut **CERTLABL** du canal. Par conséquent, lorsqu'un client tente de se connecter, le gestionnaire de files d'attente renvoie le code retour 2393 MQRC_SSL_INITIALIZATION_ERROR.

Avant d'activer le gestionnaire de files d'attente, utilisez la commande **mqcertck** pour vérifier la configuration TLS du gestionnaire de files d'attente.

Vous exécutez la commande mqcertck QM1 et recevez la sortie suivante:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
+-----------------------------------------------------------
| IBM MQ TLS Configuration Test tool
+-----------------------------------------------------------
| Problem identified:
|  No certificate could be found for the channel
|  MQCERTCK.CHANNEL
|  This tool looked in the Queue Manager's key repository
|  located at: 'C:\MQ Data\qmgrs\QM1\ssl\key.kdb'
|  for a certificate with label 'chacert',
|  which is the certificate specified in the channel's
|  CERTLABL attribute, but was unable to find one.
|
| Possible resolution:
|  A valid certificate with the label chacert
|  needs to be added to the key repository.
|
|  Alternatively, alter the channel definition to remove
|  the CERTLABL value. This can be done by executing the
|  following command in runmqsc:
|      ALTER CHANNEL(<Name>) CHLTYPE(<TYPE>) CERTLABL(' ')
+-----------------------------------------------------------
| mqcertck has ended. See above for any problems found.
| If there are problems then resolve these and run this
| tool again.
|
+-----------------------------------------------------------
```

Cette sortie vous invite à vérifier votre définition de canal pour le canal de connexion serveur MQCERTCK.CHANNEL. Ici, vous voyez l'erreur que vous avez faite et pouvez la corriger avant d'exécuter à nouveau la commande mqcertck pour vérifier que vous avez résolu le problème.

Vérification des connexions client

La commande **mqcertck** permet de vérifier les référentiels de clés client, ainsi que la configuration TLS du gestionnaire de files d'attente. Pour ce faire, **mqcertck** doit pouvoir accéder au référentiel de clés du client à partir de la machine exécutant le gestionnaire de files d'attente.

Lors de l'exécution de la commande **mqcertck** , si vous fournissez le paramètre **-clientkeyr** avec l'emplacement du référentiel de clés client (à l'exclusion de l'extension) **mqcertck** , ce référentiel de clés est vérifié par rapport au gestionnaire de files d'attente.

Si vous savez quel canal le client utilisera pour se connecter au gestionnaire de files d'attente, vous pouvez le spécifier avec l'indicateur **-clientchannel** .

Si le client utilise l'authentification mutuelle pour se connecter au gestionnaire de files d'attente, vous pouvez utiliser le paramètre **-clientusername** ou **-clientlabel** pour indiquer à la commande **mqcertck** le certificat à utiliser dans le référentiel de clés du client.

Si vous utilisez le certificat par défaut et que vous ne fournissez pas de libellé de certificat à l'application client, vous pouvez utiliser **-clientusername** et les paramètres **username** qui exécutent cette application.

Lors de l'exécution de la commande **mqcertck** , la commande génère le libellé de certificat `ibmwebspheremqXXXX` , où XXXX est la valeur transmise dans le paramètre **-clientusername** .

Afin de vérifier complètement le référentiel de clés du client, la commande **mqcertck** crée une connexion factice à l'aide de GSKit. Pour ce faire, la commande doit disposer d'un port disponible auquel elle peut se connecter lors de ses tests client. Le port par défaut utilisé est 5857. Toutefois, s'il est déjà utilisé, vous pouvez spécifier un port différent à utiliser lors des tests client.

Remarque : Bien que la commande **mqcertck** se lie à un port, aucune communication externe n'est utilisée par **mqcertck**et tous les tests sont effectués localement.

SSL/TLS sur IBM MQ MQI client

IBM MQ prend en charge TLS sur les clients. Vous pouvez personnaliser l'utilisation de TLS de différentes manières.

IBM MQ fournit une prise en charge TLS pour IBM MQ MQI clients sur les systèmes Windowset UNIX and Linux . Si vous utilisez IBM MQ classes for Java, voir Utilisation de IBM MQ classes for Java et si vous utilisez IBM MQ classes for JMS, voir Utilisation de IBM MQ classes for JMS. Le reste de cette section ne s'applique pas aux environnements Java ou JMS .

Vous pouvez spécifier le référentiel de clés pour un IBM MQ MQI client avec la valeur MQSSLKEYR dans votre fichier de configuration client IBM MQ ou lorsque votre application effectue un appel MQCONNX. Vous disposez de trois options pour spécifier qu'un canal utilise TLS:

- Utilisation d'une table de définition de canal
- Utilisation de la structure des options de configuration SSL, MQSCO, sur un appel MQCONNX
- Utilisation d' Active Directory (sur les systèmes Windows)

Vous ne pouvez pas utiliser la variable d'environnement MQSERVER pour indiquer qu'un canal utilise TLS.

Vous pouvez continuer à exécuter vos applications IBM MQ MQI client existantes sans TLS, tant que TLS n'est pas spécifié à l'autre extrémité du canal.

Si des modifications sont apportées sur une machine client au contenu du référentiel de clés TLS, à l'emplacement du référentiel de clés TLS, aux informations d'authentification ou aux paramètres matériels de cryptographie, vous devez arrêter toutes les connexions TLS afin de refléter ces modifications dans les canaux de connexion client utilisés par l'application pour se connecter au gestionnaire de files d'attente. Une fois toutes les connexions terminées, redémarrez les canaux TLS. Tous les nouveaux paramètres TLS sont utilisés. Ces paramètres sont analogues à ceux actualisés par la commande REFRESH SECURITY TYPE (SSL) sur les systèmes de gestionnaire de files d'attente.

Lorsque IBM MQ MQI client s'exécute sur un système Windows, UNIX and Linux avec du matériel de cryptographie, vous configurez ce matériel avec la variable d'environnement MQSSLCRYP. Cette variable est équivalente au paramètre SSLCRYP de la commande ALTER QMGR MQSC. Pour obtenir une description du paramètre SSLCRYP dans la commande ALTER QMGR MQSC, voir ALTER QMGR . Si vous utilisez la version GSK_PCS11 du paramètre SSLCRYP, le libellé de jeton PKCS #11 doit être indiqué en minuscules.

La réinitialisation de la clé secrète TLS et la norme FIPS sont prises en charge sur IBM MQ MQI clients. Pour plus d'informations, voir «Réinitialisation des clés secrètes SSL et TLS», à la page 460 et «FIPS (Federal Information Processing Standards) pour UNIX, Linux, and Windows», à la page 34.

Voir «Configuration de la sécurité IBM MQ MQI client», à la page 277 pour plus d'informations sur la prise en charge de TLS pour IBM MQ MQI clients.

Tâches associées

Configuration d'un client à l'aide d'un fichier de configuration

Spécification du fait qu'un canal MQI utilise SSL/TLS

Pour qu'un canal MQI utilise TLS, la valeur de l'attribut *SSLCipherSpec* du canal de connexion client doit être le nom d'un CipherSpec pris en charge par IBM MQ sur la plateforme client.

Vous pouvez définir un canal de connexion client avec une valeur pour cet attribut de l'une des manières suivantes. Ils sont répertoriés par ordre de priorité décroissante.

1. Lorsqu'un exit PreConnect fournit une structure de définition de canal à utiliser.

   Un exit PreConnect peut fournir le nom d'un CipherSpec dans la zone *SSLCipherSpec* d'une structure de définition de canal, MQCD. Cette structure est renvoyée dans la zone **ppMQCDArrayPtr** de la structure de paramètres d'exit MQNXP utilisée par l'exit PreConnect .
2. Lorsqu'une application IBM MQ MQI client émet un appel MQCONNX.

   L'application peut spécifier le nom d'un CipherSpec dans la zone *SSLCipherSpec* d'une structure de définition de canal, MQCD.Cette structure est référencée par la structure d'options de connexion, MQCNO, qui est un paramètre de l'appel MQCONNX.
3. Utilisation d'une table de définition de canal du client (CCDT).

   Une ou plusieurs entrées d'une table de définition de canal du client peuvent spécifier le nom d'un CipherSpec.Par exemple, si vous créez une entrée à l'aide de la commande MQSC DEFINE CHANNEL, vous pouvez utiliser le paramètre SSLCIPH dans la commande pour spécifier le nom d'un CipherSpec.
4. Utilisation de Active Directory sous Windows.

   Sur les systèmes Windows , vous pouvez utiliser la commande de contrôle **setmqscp** pour publier les définitions de canal de connexion client dans Active Directory. Une ou plusieurs de ces définitions peuvent spécifier le nom d'un CipherSpec.

Par exemple, si une application client fournit une définition de canal de connexion client dans une structure MQCD sur un appel MQCONNX, cette définition est utilisée de préférence aux entrées d'une table de définition de canal du client accessibles par le client IBM MQ .

Vous ne pouvez pas utiliser la variable d'environnement MQSERVER pour fournir la définition de canal à l'extrémité client d'un canal MQI qui utilise TLS.

Pour vérifier si un certificat client a transité, affichez le statut du canal à l'extrémité serveur d'un canal pour la présence d'une valeur de paramètre de nom d'homologue.

Concepts associés

«Spécification d'un CipherSpec pour un IBM MQ MQI client», à la page 449
Vous disposez de trois options pour spécifier un CipherSpec pour un IBM MQ MQI client.

CipherSpecs et CipherSuites dans IBM MQ

IBM MQ prend en charge TLS 1.2 CipherSpecset les algorithmes RSA et Diffie-Hellman. Toutefois, vous pouvez activer les CipherSpecsobsolètes, si vous devez le faire.

Voir «Activation des CipherSpecs», à la page 434 pour plus d'informations sur:

- CipherSpecs pris en charge par IBM MQ.
- Comment activer les CipherSpecs SSL 3.0 et TLS 1.0 CipherSpecs.

IBM MQ prend en charge les algorithmes d'authentification et d'échange de clés RSA et Diffie-Hellman. La taille de la clé utilisée lors de l'établissement de liaison TLS peut dépendre du certificat numérique que vous utilisez, mais certains CipherSpecs incluent une spécification de la taille de la clé d'établissement de liaison. Plus la taille de clé est élevée, plus l'authentification est solide. Avec des tailles de clé plus petites, l'établissement de la liaison est plus rapide.

Concepts associés

«CipherSpecs et CipherSuites», à la page 19
Les protocoles de sécurité cryptographique doivent convenir des algorithmes utilisés par une connexion sécurisée. CipherSpecs et CipherSuites définissent des combinaisons spécifiques d'algorithmes.

NSA Suite B Cryptography dans IBM MQ

Cette rubrique explique comment configurer IBM MQ sous Windows, Linuxet UNIX pour qu'il soit conforme au profil TLS 1.2 conforme à la norme Suite B.

Au fil du temps, la norme NSA Cryptography Suite B est mise à jour pour refléter les nouvelles attaques contre les algorithmes et les protocoles de chiffrement. Par exemple, certains CipherSpecs peuvent ne plus être certifiés Suite B. Lorsque de telles modifications se produisent, IBM MQ est également mis à jour pour implémenter la dernière norme. Vous pouvez alors constater des changements de comportement une fois la maintenance appliquée. Le fichier Readme IBM MQ répertorie la version de Suite B appliquée par chaque niveau de maintenance du produit. Si vous configurez IBM MQ pour appliquer la conformité Suite B, consultez toujours le fichier Readme lors de la planification de l'application de la maintenance. Voir Fichiers Readme des produits IBM MQ, WebSphere MQet MQSeries.

Sur les systèmes Windows, UNIXet Linux , IBM MQ peut être configuré pour se conformer au profil TLS 1.2 compatible Suite B aux niveaux de sécurité indiqués dans le tableau 1.

Tableau 3. Niveaux de sécurité Suite B avec CipherSpecs et algorithmes de signature numérique autorisés

Niveau de sécurité	**CipherSpecs autorisés**	**Algorithmes de signature numérique autorisés**
128 bits	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA avec SHA-256 ECDSA avec SHA-384
192 bits	ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA avec SHA-384
Les deux [1]	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA avec SHA-256 ECDSA avec SHA-384

1. Il est possible de configurer simultanément les niveaux de sécurité 128 bits et 192 bits. Etant donné que la configuration Suite B détermine les algorithmes de cryptographie minimaux acceptables, la configuration des deux niveaux de sécurité est équivalente à la configuration du niveau de sécurité 128 bits uniquement. Les algorithmes de cryptographie du niveau de sécurité 192 bits sont plus forts que le minimum requis pour le niveau de sécurité 128 bits, de sorte qu'ils sont autorisés pour le niveau de sécurité 128 bits même si le niveau de sécurité 192 bits n'est pas activé.

Remarque : Les conventions de dénomination utilisées pour le niveau de sécurité ne représentent pas nécessairement la taille de courbe elliptique ou la taille de clé de l'algorithme de chiffrement AES.

CipherSpec -conformation vers Suite B

Bien que le comportement par défaut de IBM MQ ne soit pas conforme à la norme Suite B, IBM MQ peut être configuré pour être conforme à l'un des niveaux de sécurité, ou aux deux, sur les systèmes UNIX and Linux Windows. Suite à la configuration réussie de IBM MQ pour utiliser Suite B, toute tentative de démarrage d'un canal sortant à l'aide d'un CipherSpec non conforme à Suite B entraîne l'erreur AMQ9282. Cette activité a également pour conséquence que le client MQI renvoie le code anomalie MQRC_CIPHER_SPEC_NOT_SUITE_B. De même, la tentative de démarrage d'un canal entrant à l'aide d'un CipherSpec non conforme à la configuration Suite B entraîne l'erreur AMQ9616.

Pour plus d'informations sur les CipherSpecs IBM MQ CipherSpecs, voir «Activation des CipherSpecs», à la page 434

Suite B et certificats numériques

La suite B restreint les algorithmes de signature numérique qui peuvent être utilisés pour signer des certificats numériques. La suite B restreint également le type de clé publique que les certificats peuvent contenir. Par conséquent, IBM MQ doit être configuré pour utiliser des certificats dont l'algorithme de signature numérique et le type de clé publique sont autorisés par le niveau de sécurité Suite B configuré du partenaire distant. Les certificats numériques qui ne sont pas conformes aux exigences de niveau de sécurité sont rejetés et la connexion échoue avec l'erreur AMQ9633 ou AMQ9285.

Pour le niveau de sécurité Suite B 128 bits, la clé publique du sujet de certificat doit utiliser la courbe elliptique NIST P-256 ou la courbe elliptique NIST P-384 et être signée avec la courbe elliptique NIST P-256 ou la courbe elliptique NIST P-384 . Au niveau de la sécurité Suite B 192 bits, la clé publique du sujet de certificat est requise pour utiliser la courbe elliptique NIST P-384 et pour être signée avec la courbe elliptique NIST P-384 .

Pour obtenir un certificat adapté à une opération conforme à la suite B, utilisez la commande **runmqakm** et spécifiez le paramètre **-sig_alg** pour demander un algorithme de signature numérique approprié. Les valeurs des paramètres EC_ecdsa_with_SHA256 et EC_ecdsa_with_SHA384 **-sig_alg** correspondent à des clés de courbe elliptique signées par les algorithmes de signature numérique Suite B autorisés.

Pour plus d'informations sur la commande **runmqakm** , voir Options runmqckm et runmqakm.

Remarque : Les commandes **runmqckm** et **strmqikm** ne prennent pas en charge la création de certificats numériques pour une opération compatible avec Suite B.

Création et demande de certificats numériques

Pour créer un certificat numérique autosigné pour les tests Suite B, voir «Création d'un certificat personnel autosigné sur UNIX, Linux, and Windows», à la page 303

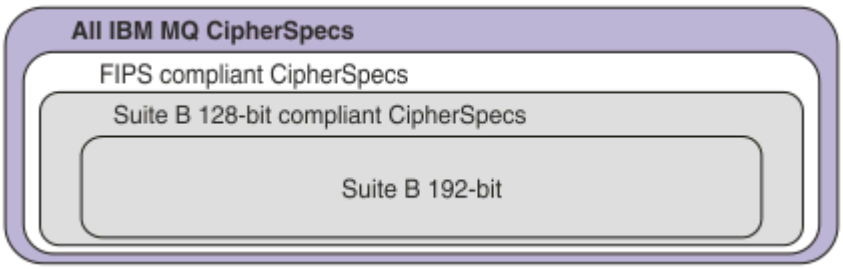
Pour demander un certificat numérique signé par une autorité de certification pour une utilisation en production Suite B, voir «Demande d'un certificat personnel sur UNIX, Linux, and Windows», à la page 306.

Remarque : L'autorité de certification utilisée doit générer des certificats numériques qui répondent aux exigences décrites dans le document IETF RFC 6460.

FIPS 140-2 et Suite B

La norme Suite B est conceptuellement similaire à la norme FIPS 140-2, car elle restreint l'ensemble des algorithmes de cryptographie activés afin de fournir un niveau de sécurité assuré. Les CipherSpecs Suite B actuellement pris en charge peuvent être utilisés lorsque IBM MQ est configuré pour une opération conforme à la norme FIPS 140-2. Il est donc possible de configurer IBM MQ pour la conformité FIPS et Suite B simultanément, auquel cas les deux ensembles de restrictions s'appliquent.

Le diagramme suivant illustre la relation entre ces sous-ensembles:



Configuration de IBM MQ pour une opération compatible Suite B

Pour plus d'informations sur la configuration de IBM MQ sur Windows, UNIX and Linux pour une opération compatible avec Suite B, voir «Configuration de IBM MQ pour Suite B», à la page 43.

IBM MQ ne prend pas en charge les opérations conformes à la suite B sur les plateformes IBM i et z/OS . Les clients IBM MQ Java et JMS ne prennent pas non plus en charge les opérations compatibles avec Suite B.

Concepts associés

«Comment indiquer que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur MQI Client», à la page 278
Créez vos référentiels de clés à l'aide d'un logiciel compatible FIPS, puis indiquez que le canal doit utiliser des CipherSpecscertifiés FIPS.

Configuration de IBM MQ pour Suite B

IBM MQ peut être configuré pour fonctionner en conformité avec la norme NSA Suite B sur les plateformes Windows, UNIX and Linux .

La suite B restreint l'ensemble des algorithmes de cryptographie activés afin de fournir un niveau de sécurité assuré. IBM MQ peut être configuré pour fonctionner conformément à la Suite B afin de fournir un niveau de sécurité amélioré. Pour plus d'informations sur la suite B, voir «Agence de sécurité nationale (NSA) Suite B Cryptographie», à la page 21. Pour plus d'informations sur la configuration de la suite B et son effet sur les canaux TLS, voir «NSA Suite B Cryptography dans IBM MQ», à la page 41.

Gestionnaire de files d'attente

Pour un gestionnaire de files d'attente, utilisez la commande **ALTER QMGR** avec le paramètre **SUITEB** pour définir les valeurs appropriées à votre niveau de sécurité requis. Pour plus d'informations, voir ALTER QMGR.

Vous pouvez également utiliser la commande PCF **MQCMD_CHANGE_Q_MGR** avec le paramètre **MQIA_SUITE_B_STRENGTH** pour configurer le gestionnaire de files d'attente pour une opération compatible avec Suite B.

Remarque : Si vous modifiez les paramètres Suite B d'un gestionnaire de files d'attente, vous devez redémarrer le service MQXR pour que ces paramètres prennent effet.

MQI Client

Par défaut, les clients MQI n'appliquent pas la conformité Suite B. Vous pouvez activer le client MQI pour la conformité Suite B en exécutant l'une des options suivantes:

1. En définissant la zone **EncryptionPolicySuiteB** dans la structure MQSCO d'un appel MQCONNX sur une ou plusieurs des valeurs suivantes:
   - MQ_SUITE_B_NONE
   - MQ_SUITE_B_128_BIT
   - MQ_SUITE_B_192_BIT

   L'utilisation de MQ_SUITE_B_NONE avec une autre valeur n'est pas valide.
2. En définissant la variable d'environnement MQSUITEB sur une ou plusieurs des valeurs suivantes:
   - Aucun
   - 128_BIT
   - 192_BIT

   Vous pouvez spécifier plusieurs valeurs à l'aide d'une liste séparée par des virgules. L'utilisation de la valeur NONE avec une autre valeur n'est pas valide.
3. En définissant l'attribut **EncryptionPolicySuiteB** dans la strophe SSL du fichier de configuration du client MQI sur une ou plusieurs des valeurs suivantes:
   - Aucun
   - 128_BIT
   - 192_BIT

Vous pouvez spécifier plusieurs valeurs à l'aide d'une liste séparée par des virgules. L'utilisation de NONE avec une autre valeur n'est pas valide.

Remarque : Les paramètres du client MQI sont répertoriés par ordre de priorité. La structure MSCO de l'appel MQCONNX remplace le paramètre de la variable d'environnement MQSUITEB, qui remplace l'attribut dans la strophe SSL.

Pour plus de détails sur la structure MQSCO, voir MQSCO-Options de configuration SSL.

Pour plus d'informations sur l'utilisation de Suite B dans le fichier de configuration du client, voir Strophe SSL du fichier de configuration du client.

Pour plus d'informations sur l'utilisation de la variable d'environnement MQSUITEB, voir Description des variables d'environnement.

.NET

Pour les clients .NET non gérés, la propriété **MQC.ENCRYPTION_POLICY_SUITE_B** indique le type de sécurité Suite B requis.

Pour plus d'informations sur l'utilisation de la suite B dans IBM MQ classes for .NET, voir Classe MQEnvironment .NET.

AMQP

Les paramètres d'attribut Suite B d'un gestionnaire de files d'attente s'appliquent aux canaux AMQP de ce gestionnaire de files d'attente. Si vous modifiez les paramètres de la suite de gestionnaires de files d'attente B, vous devez redémarrer le service AMQP pour que les modifications soient prises en compte.

Règles de validation de certificat dans IBM MQ

La règle de validation de certificat détermine dans quelle mesure la validation de la chaîne de certificats est conforme aux normes de sécurité de l'industrie.

La règle de validation de certificat dépend de la plateforme et de l'environnement comme suit:

- Pour les applications Java et JMS sur toutes les plateformes, la règle de validation de certificat dépend du composant JSSE de l'environnement d'exécution Java . Pour plus d'informations sur les règles de validation de certificat, voir la documentation de votre environnement d'exécution Java.
- Pour les systèmes IBM i , la règle de validation de certificat dépend de la bibliothèque de sockets sécurisés fournie par le système d'exploitation. Pour plus d'informations sur les règles de validation de certificat, voir la documentation du système d'exploitation.
- Pour les systèmes z/OS , la règle de validation de certificat dépend du composant System SSL fourni par le système d'exploitation. Pour plus d'informations sur les règles de validation de certificat, voir la documentation du système d'exploitation.
- Pour les systèmes UNIX, Linux, and Windows , la règle de validation de certificat est fournie par GSKit et peut être configurée. Deux règles de validation de certificat différentes sont prises en charge:
  - Une règle de validation de certificat existante, utilisée pour une compatibilité et une interopérabilité maximales en amont avec les anciens certificats numériques qui ne sont pas conformes aux normes de validation de certificat IETF actuelles. Cette règle est appelée règle de base.
  - Une stratégie de validation de certificat stricte et conforme aux normes qui applique la norme RFC 5280. Cette règle est connue sous le nom de règle standard.

Pour plus d'informations sur la configuration de la règle de validation de certificat sous UNIX, Linux, and Windows, voir «Configuration des règles de validation de certificat dans IBM MQ», à la page 44. Pour plus d'informations sur les différences entre les règles de validation de certificat de base et standard, voir Certificate validation and trust policy design on UNIX, Linux, and Windows.

Configuration des règles de validation de certificat dans IBM MQ

Vous pouvez spécifier les règles de validation de certificat TLS à utiliser pour valider les certificats numériques reçus des systèmes partenaires distants de quatre manières.

Sur le gestionnaire de files d'attente, la règle de validation de certificat peut être définie comme suit:

- Utilisation de l'attribut de gestionnaire de files d'attente *CERTVPOL*. Pour plus d'informations sur la définition de cet attribut, voir ALTER QMGR.

Sur le client, plusieurs méthodes peuvent être utilisées pour définir la règle de validation de certificat. Si plusieurs méthodes sont utilisées pour définir la règle, le client utilise les paramètres dans l'ordre de priorité suivant:

1. Utilisation de la zone *CertificateValPolicy* dans la structure MQSCO du client. Pour plus d'informations sur l'utilisation de cette zone, voir MQSCO-Options de configuration SSL.
2. Utilisation de la variable d'environnement client *MQCERTVPOL*. Pour plus d'informations sur l'utilisation de cette variable, voir MQCERTVPOL.
3. Utilisation de la valeur du paramètre d'optimisation de la section SSL du client, *CertificateValPolicy*. Pour plus d'informations sur l'utilisation de ce paramètre, voir Strophe SSL du fichier de configuration du client.

Pour plus d'informations sur les règles de validation de certificat, voir «Règles de validation de certificat dans IBM MQ», à la page 44.

Certificats numériques et compatibilité CipherSpec dans IBM MQ

Cette rubrique fournit des informations sur la façon de choisir les CipherSpecs et les certificats numériques appropriés pour votre règle de sécurité, en soulignant la relation entre les CipherSpecs et les certificats numériques dans IBM MQ.

Seul un sous-ensemble des CipherSpecs pris en charge peut être utilisé avec tous les types de certificat numérique pris en charge. Il est donc nécessaire de choisir un CipherSpec approprié pour votre certificat numérique. De même, si la stratégie de sécurité de votre organisation requiert que vous utilisiez un CipherSpec particulier, vous devez obtenir un certificat numérique approprié pour ce CipherSpec.

L'algorithme de signature numérique MD5 et TLS 1.2

Les certificats numériques signés à l'aide de l'algorithme MD5 sont rejetés lorsque le protocole TLS 1.2 est utilisé. En effet, l'algorithme MD5 est désormais considéré comme faible par de nombreux analystes cryptographiques et son utilisation est généralement déconseillée. Pour utiliser des CipherSpecs plus récents basés sur le protocole TLS 1.2 , assurez-vous que les certificats numériques n'utilisent pas l'algorithme MD5 dans leurs signatures numériques. Les CipherSpecs plus anciens qui utilisent les protocoles TLS 1.0 ne sont pas soumis à cette restriction et peuvent continuer à utiliser des certificats avec des signatures numériques MD5 .

Pour afficher l'algorithme de signature numérique d'un certificat particulier, vous pouvez utiliser la commande **runmqakm** :

```
runmqakm -cert -details -db key.kdb -pw password -label cert_label
```

où *cert_label* est le libellé de certificat de l'algorithme de signature numérique à afficher. Pour plus de détails voir Labels de certificat numérique.

Remarque : Bien que l'interface graphique **runmqckm** (iKeycmd) et **strmqikm** (iKeyman) puissent être utilisées pour afficher une sélection d'algorithmes de signature numérique, l'outil **runmqakm** offre une gamme plus large.

L'exécution de la commande **runmqakm** génère une sortie affichant l'utilisation de l'algorithme de signature spécifié:

```
Label : ibmmqexample
Key Size : 1024
Version : X509 V3
Serial : 4e4e93f1
Issuer : CN=Old Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : August 19, 2011 5:48:49 PM GMT+01:00
```

```
Not After : August 18, 2012 5:48:49 PM GMT+01:00
Public Key
    30 81 9F 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01
    05 00 03 81 8D 00 30 81 89 02 81 81 00 98 5A 7A
    F0 18 21 EE E4 8A 6E DE C8 01 4B 3A 1E 41 90 3D
    CE 01 3F E6 32 30 6C 23 59 F0 FE 78 6D C2 80 EF
    BC 83 54 7A EB 60 80 62 6B F1 52 FE 51 9D C1 61
    80 A5 1C D4 F0 76 C7 15 6D 1F 0D 4D 31 3E DC C6
    A9 20 84 6E 14 A1 46 7D 4C F5 79 4D 37 54 0A 3B
    A9 74 ED E7 8B 0F 80 31 63 1A 0B 20 A5 99 EE 0A
    30 A6 B6 8F 03 97 F6 99 DB 6A 58 89 7F 27 34 DE
    55 08 29 D8 A9 6B 46 E6 02 17 C3 13 D3 02 03 01
    00 01
Public Key Type : RSA (1.2.840.113549.1.1.1)
Fingerprint : SHA1 :
    09 4E 4F F2 1B CB C1 F4 4F 15 C9 2A F7 32 0A 82
    DA 45 92 9F
Fingerprint : MD5 :
    44 54 81 7C 58 68 08 3A 5D 75 96 40 D5 8C 7A CB
Fingerprint : SHA256 :
    3B 47 C6 E7 7B B0 FF 85 34 E7 48 BE 11 F2 D4 35
    B7 9A 79 53 2B 07 F5 E7 65 E8 F7 84 E0 2E 82 55
Signature Algorithm : MD5WithRSASignature (1.2.840.113549.1.1.4)
Value
    3B B9 56 E6 F2 77 94 69 5B 3F 17 EA 7B 19 D0 A2
    D7 10 38 F1 88 A4 44 1B 92 35 6F 3B ED 99 9B 3A
    A5 A4 FC 72 25 5A A9 E3 B1 96 88 FC 1E 9F 9B F1
    C5 E8 8E CF C4 8F 48 7B 0E A6 BB 13 AE 2B BD D8
    63 2C 03 38 EF DC 01 E1 1F 7A 6F FB 2F 65 74 D0
    FD 99 94 BA B2 3A D5 B4 89 6C C1 2B 43 6D E2 39
    66 6A 65 CB C3 C4 E2 CC F5 49 39 A3 8B 93 5A DD
    B0 21 0B A8 B2 59 5B 24 59 50 44 89 DC 78 19 51
Trust Status : Enabled
```

La ligne `Signature Algorithm` indique que l'algorithme `MD5WithRSASignature` est utilisé. Cet algorithme étant basé sur MD5 , ce certificat numérique ne peut pas être utilisé avec les CipherSpecsTLS 1.2 .

Interopérabilité de Elliptic Curve et de RSA CipherSpecs

V 9.1.4 Les CipherSpecs ne peuvent pas tous être utilisés avec tous les certificats numériques. CipherSpecs sont indiqués par le préfixe de nom CipherSpec . Chaque type de CipherSpec impose des restrictions différentes sur le type de certificat numérique qui peut être utilisé. Ces restrictions s'appliquent à toutes les connexions TLS IBM MQ , mais sont particulièrement pertinentes pour les utilisateurs de la cryptographie Elliptic Curve.

Le tableau suivant récapitule les relations entre les CipherSpecs et les certificats numériques:

Tableau 4. Relations entre les CipherSpecs et les certificats numériques

Tapez	CipherSpec Préfixe de nom	Description	Type de clé publique requis	Algorithme de chiffrement de signature numérique	Méthode d'établissement de clé secrète
1	ECDHE_ECDSA_	CipherSpecs qui utilisent des clés publiques Elliptic Curve, des clés secrètes Elliptic Curve et des algorithmes de signature numérique Elliptic Curve.	Elliptic Curve	ECDSA	ECDHE
2	ECDHE_RSA_	CipherSpecs qui utilisent des clés publiques RSA, des clés secrètes Elliptic Curve et des algorithmes de signature numérique RSA.	RSA	RSA	ECDHE

Tableau 4. Relations entre les CipherSpecs et les certificats numériques (suite)					
Tapez	**CipherSpec Préfixe de nom**	**Description**	**Type de clé publique requis**	**Algorithme de chiffrement de signature numérique**	**Méthode d'établissement de clé secrète**
V 9.1.4 3	(Tous les CipherSpecs TLS 1.3 CipherSpecs)	CipherSpecs qui utilisent des clés publiques Elliptic Curve ou RSA, des clés secrètes Elliptic Curve et des algorithmes de signature numérique Elliptic Curve ou RSA.	Courbe elliptique ou RSA	ECDSA ou RSA	ECDHE ou RSA
4	(Tous les autres)	CipherSpecs qui utilisent des clés publiques RSA et des algorithmes de signature numérique RSA.	RSA	RSA	RSA

Remarque : Les CipherSpecs de type 1 et 2 ne sont pas pris en charge par les gestionnaires de files d'attente IBM MQ et les clients MQI sur la plateforme IBM i .

La colonne de type de clé publique obligatoire indique le type de clé publique que le certificat personnel doit posséder lors de l'utilisation de chaque type de CipherSpec. Le certificat personnel est le certificat d'entité finale qui identifie le gestionnaire de files d'attente ou le client auprès de son partenaire distant.

Vous pouvez configurer un canal avec un CipherSpec qui requiert un certificat Elliptic Curve (EC) et un libellé de certificat pour un certificat RSA, ou l'inverse. Vous devez vous assurer que le certificat nommé dans le libellé de certificat est approprié pour le canal CipherSpec.

En supposant que vous avez correctement configuré IBM MQ, vous pouvez avoir:

- Un seul gestionnaire de files d'attente avec un mélange de certificats RSA et EC.
- Différents canaux sur le même gestionnaire de files d'attente à l'aide d'un certificat RSA ou EC.

L'algorithme de chiffrement de signature numérique fait référence à l'algorithme de chiffrement utilisé pour valider l'homologue. L'algorithme de chiffrement est utilisé avec un algorithme de hachage tel que MD5, SHA-1 ou SHA-256 pour calculer la signature numérique. Il existe différents algorithmes de signature numérique qui peuvent être utilisés, par exemple RSA avec MD5 ou ECDSA avec SHA-256. Dans le tableau, ECDSA fait référence à l'ensemble des algorithmes de signature numérique qui utilisent ECDSA ; RSA fait référence à l'ensemble des algorithmes de signature numérique qui utilisent RSA. Tout algorithme de signature numérique pris en charge dans l'ensemble peut être utilisé, à condition qu'il soit basé sur l'algorithme de chiffrement indiqué.

Les CipherSpecs de type 1 requièrent que le certificat personnel ait une clé publique Elliptic Curve. Lorsque ces CipherSpecs sont utilisés, l'accord de clé éphémère Elliptic Curve Diffie Hellman est utilisé pour établir la clé secrète pour la connexion.

Les CipherSpecs de type 2 requièrent que le certificat personnel ait une clé publique RSA. Lorsque ces CipherSpecs sont utilisés, l'accord de clé éphémère Elliptic Curve Diffie Hellman est utilisé pour établir la clé secrète pour la connexion.

Les CipherSpecs de type 3 requièrent que le certificat personnel ait une clé publique RSA. Lorsque ces CipherSpecs sont utilisés, l'échange de clés RSA est utilisé pour établir la clé secrète pour la connexion.

Cette liste de restrictions n'est pas exhaustive: selon la configuration, il peut y avoir des restrictions supplémentaires qui peuvent affecter davantage la possibilité d'interopérer. Par exemple, si IBM MQ est configuré pour être conforme aux normes FIPS 140-2 ou NSA Suite B, cela limitera également la plage des configurations autorisées. Pour plus d'informations, reportez-vous à la section suivante.

Si vous devez utiliser différents types de CipherSpec sur le même gestionnaire de files d'attente ou la même application client, configurez un libellé de certificat approprié et une combinaison de CipherSpec sur la définition du client.

Les trois types de CipherSpec n'interopèrent pas directement: il s'agit d'une limitation des normes TLS actuelles. Par exemple, supposons que vous ayez choisi d'utiliser le CipherSpec ECDHE_ECDSA_AES_128_CBC_SHA256 pour un canal récepteur nommé TO.QM1 sur un gestionnaire de files d'attente nommé QM1, le récepteur doit alors disposer d'un certificat personnel avec une clé Elliptic Curve et une signature numérique basée sur ECDSA. Si le canal récepteur ne répond pas à ces exigences, le démarrage du canal échoue.

Les autres canaux se connectant au gestionnaire de files d'attente QM1 peuvent utiliser d'autres CipherSpecs, à condition que chaque canal utilise un certificat du type approprié pour le CipherSpec de ce canal. Par exemple, supposons que QM1 utilise un canal émetteur nommé TO.QM2 pour envoyer des messages à un autre gestionnaire de files d'attente nommé QM2. Canal TO.QM2 peut utiliser le type 3 CipherSpec TLS_RSA_WITH_AES_256_CBC_SHA256 à condition que les deux extrémités du canal utilisent des certificats contenant des clés publiques RSA. L'attribut de canal de label de certificat peut être utilisé pour configurer un certificat différent pour chaque canal.

Lors de la planification de vos réseaux IBM MQ , réfléchissez soigneusement aux canaux qui requièrent TLS et assurez-vous que le type de certificat utilisé pour chaque canal est approprié pour une utilisation avec le CipherSpec sur ce canal.

Pour afficher l'algorithme de signature numérique et le type de clé publique d'un certificat numérique, vous pouvez utiliser la commande **runmqakm** :

```
runmqakm -cert -details -db key.kdb -pw password -label cert_label
```

où *cert_label* est le libellé du certificat dont vous devez afficher l'algorithme de signature numérique. Pour plus de détails voir Labels de certificat numérique.

L'exécution de la commande **runmqakm** génère une sortie affichant le type de clé publique:

```
Label : ibmmqexample
Key Size : 384
Version : X509 V3
Serial : 9ad5eeef5d756f41
Issuer : CN=Example Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : 21 August 2011 13:10:24 GMT+01:00
Not After : 21 August 2012 13:10:24 GMT+01:00
Public Key
    30 76 30 10 06 07 2A 86 48 CE 3D 02 01 06 05 2B
    81 04 00 22 03 62 00 04 3E 6F A9 06 B6 C3 A0 11
    F8 D6 22 78 FE EF 0A FE 34 52 C0 8E AB 5E 81 73
    D0 97 3B AB D6 80 08 E7 31 E9 18 3F 6B DE 06 A7
    15 D6 9D 5B 6F 56 3B 7F 72 BB 6F 1E C9 45 1C 46
    60 BE F2 DC 1B AD AC EC 64 4C 0E 06 65 6E ED 93
    B8 F5 95 E0 F9 2A 05 D6 21 02 BD FB 06 63 A1 CC
    66 C6 8A 0A 5C 3F F7 D3
Public Key Type : EC_ecPublicKey (1.2.840.10045.2.1)
Fingerprint : SHA1 :
    3C 34 58 04 5B 63 5F 5C C9 7A E7 67 08 2B 84 43
    3D 43 7A 79
Fingerprint : MD5 :
    49 13 13 E1 B2 AC 18 9A 31 41 DC 8C B4 D6 06 68
Fingerprint : SHA256 :
    6F 76 78 68 F3 70 F1 53 CE 39 31 D9 05 C5 C5 9F
    F2 B8 EE 21 49 16 1D 90 64 6D AC EB 0C A7 74 17
Signature Algorithm : EC_ecdsa_with_SHA384 (1.2.840.10045.4.3.3)
Value
    30 65 02 30 0A B0 2F 72 39 9E 24 5A 22 FE AC 95
    0D 0C 6D 6C 2F B3 E7 81 F6 C1 36 1B 9A B0 6F 07
    59 2A A1 4C 02 13 7E DD 06 D6 FE 4B E4 03 BC B1
    AC 49 54 1E 02 31 00 90 0E 46 2B 04 37 EE 2C 5F
    1B 9C 69 E5 99 60 84 84 10 71 1A DA 63 88 33 E2
    22 CC E6 1A 4E F4 61 CC 51 F9 EE A0 8E F4 DC B5
    0B B9 72 58 C3 C7 A4
Trust Status : Enabled
```

Dans ce cas, la ligne Type de clé publique indique que le certificat possède une clé publique Elliptic Curve. Dans ce cas, la ligne Algorithme de signature indique que l'algorithme EC_ecdsa_with_SHA384 est en cours d'utilisation: il est basé sur l'algorithme ECDSA. Par conséquent, ce certificat ne peut être utilisé qu'avec des CipherSpecsde type 1.

Vous pouvez également utiliser la commande **runmqckm** avec les mêmes paramètres. L'interface graphique **strmqikm** peut également être utilisée pour afficher les algorithmes de signature numérique si vous ouvrez le référentiel de clés et que vous cliquez deux fois sur le libellé du certificat. Toutefois, vous devez utiliser l'outil **runmqakm** pour afficher les certificats numériques car il prend en charge un plus large éventail d'algorithmes.

TLS 1.3 CipherSpecs

V 9.1.4

TLS 1.3 CipherSpecs prend en charge les certificats ECDSA et RSA.

Elliptic Curve CipherSpecs et NSA Suite B

Lorsque IBM MQ est configuré pour se conformer au profil TLS 1.2 conforme à la suite B, les CipherSpecs et les algorithmes de signature numérique autorisés sont restreints, comme décrit dans «NSA Suite B Cryptography dans IBM MQ», à la page 41. De plus, la plage de clés Elliptic Curve acceptables est réduite en fonction des niveaux de sécurité configurés.

Au niveau de la sécurité Suite B 128 bits, la clé publique du sujet de certificat est requise pour utiliser la courbe elliptique NIST P-256 ou NIST P-384 et pour être signée avec la courbe elliptique NIST P-256 ou la courbe elliptique NIST P-384 . La commande **runmqakm** peut être utilisée pour demander des certificats numériques pour ce niveau de sécurité à l'aide d'un paramètre -sig_alg de EC_ecdsa_with_SHA256ou de EC_ecdsa_with_SHA384.

Au niveau de la sécurité de la suite B 192 bits, la clé publique du sujet de certificat est requise pour utiliser la courbe elliptique NIST P-384 et pour être signée avec la courbe elliptique NIST P-384 . La commande **runmqakm** peut être utilisée pour demander des certificats numériques pour ce niveau de sécurité à l'aide d'un paramètre -sig_alg de EC_ecdsa_with_SHA384.

Les courbes elliptiques NIST prises en charge sont les suivantes:

Tableau 5. Courbes elliptiques NIST prises en charge

Nom de courbe NIST FIPS 186-3	**Nom de courbe RFC 4492**	**Taille de clé de courbe elliptique (bits)**
P-256	secp256r1	256
P-384	secp384r1	384
P-521	secp521r1	521

Remarque : La courbe elliptique P-521 du NIST ne peut pas être utilisée pour une opération conforme à la suite B.

Concepts associés

«Activation des CipherSpecs», à la page 434
Activez un CipherSpec à l'aide du paramètre **SSLCIPH** dans la commande **DEFINE CHANNEL** MQSC ou dans la commande **ALTER CHANNEL** MQSC.

«Comment indiquer que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur MQI Client», à la page 278
Créez vos référentiels de clés à l'aide d'un logiciel compatible FIPS, puis indiquez que le canal doit utiliser des CipherSpecscertifiés FIPS.

«NSA Suite B Cryptography dans IBM MQ», à la page 41
Cette rubrique explique comment configurer IBM MQ sous Windows, Linuxet UNIX pour qu'il soit conforme au profil TLS 1.2 conforme à la norme Suite B.

«Agence de sécurité nationale (NSA) Suite B Cryptographie», à la page 21
Le gouvernement des États-Unis d'Amérique fournit des conseils techniques sur les systèmes informatiques et la sécurité, y compris le chiffrement des données. La National Security Agency (NSA) des États-Unis recommande un ensemble d'algorithmes cryptographiques interopérables dans sa norme Suite B.

Enregistrements d'authentification de canal

Pour exercer un contrôle plus précis sur les accès accordés aux systèmes en cours de connexion au niveau d'un canal, vous pouvez utiliser les enregistrements d'authentification de canal.

Vous découvrirez peut-être que des clients essaient de se connecter à votre gestionnaire de files d'attente à l'aide d'un ID utilisateur vide ou d'un ID utilisateur de niveau supérieur, permettant à un client de procéder à des actions indésirables. Vous pouvez bloquer l'accès à ces clients à l'aide d'enregistrements d'authentification de canal. Il est également possible qu'un client accepte un ID utilisateur valide sur la plateforme client, mais inconnu ou sous un format non valide sur la plateforme serveur. Vous pouvez utiliser un enregistrement d'authentification de canal pour associer l'ID utilisateur accepté à un ID utilisateur valide.

Vous pouvez trouver une application client qui se connecte à votre gestionnaire de files d'attente et adopte un comportement indésirable. Pour protéger le serveur des problèmes que cette application pourrait provoquer, il convient de bloquer temporairement l'utilisation de l'adresse IP sur laquelle se trouve l'application client, le temps de mettre à jour les règles du pare-feu ou de corriger l'application. Vous pouvez utiliser un enregistrement d'authentification de canal pour bloquer l'adresse IP à partir de laquelle l'application client se connecte.

Si vous avez défini un outil d'administration tel qu'IBM MQ Explorer et un canal pour cette utilisation particulière, il est conseillé de limiter son utilisation à des ordinateurs client spécifiques. Vous pouvez utiliser un enregistrement d'authentification de canal pour permettre l'utilisation de ce canal uniquement à partir de certaines adresses IP.

Si vous venez de commencer avec des exemples d'applications s'exécutant en tant que clients, voir Préparation et exécution des exemples de programmes pour un exemple de configuration du gestionnaire de files d'attente en toute sécurité à l'aide d'enregistrements d'authentification de canal.

Pour obtenir des enregistrements d'authentification de canal afin de contrôler les canaux de communications entrantes, utilisez la commande MQSC **ALTER QMGR CHLAUTH(ENABLED)**.

Des règles **CHLAUTH** sont appliquées à un agent MCA de canal qui est créé en réponse à une nouvelle connexion entrante. Pour un agent MCA de canal créé en réponse au démarrage du canal en local, aucune règle **CHLAUTH** n'est appliquée.

Tableau 6. Emplacement de l'application des règles CHLAUTH dans différentes paires de canaux

Type de canal	**Agent MCA sur lequel les règles CHLAUTH sont appliquées**
Emetteur-récepteur	RCVR
Demandeur-serveur (démarré sur le serveur)	RQSTR
Demandeur-serveur (démarré sur le demandeur)	SVR
Demandeur-émetteur (démarré sur l'émetteur)	RQSTR
Demandeur-émetteur (démarré sur le demandeur)	Emetteur pour la connexion initiale. Demandeur pour la connexion de rappel.

Les enregistrements d'authentification de canal peuvent être créés pour l'exécution des fonctions suivantes :

- Bloquer les connexion en provenance d'adresses IP spécifiques.
- Bloquer les connexions associées à des ID utilisateur spécifiques.

- Définir une valeur MCAUSER à utiliser pour n'importe quel canal se connectant à partir d'une adresse IP spécifique.
- Définir une valeur MCAUSER à utiliser pour n'importe quel canal acceptant un ID utilisateur spécifique.
- Définir une valeur MCAUSER à utiliser pour n'importe quel canal associé à une adresse SSL ou un nom distinctif TLS spécifique.
- Définir une valeur MCAUSER à utiliser pour n'importe quel canal se connectant à partir d'un gestionnaire de files d'attente spécifique.
- Bloquer les connexions qui prétendent provenir d'un certain gestionnaire de files d'attente sauf si la connexion provient d'une adresse IP spécifique.
- Bloquer les connexions présentant un certain certificat SSL ou TLS, sauf si la connexion provient d'une adresse IP spécifique.

Ces utilisations sont expliquées plus en détails dans les sections suivantes.

Vous pouvez créer, modifier ou supprimer des enregistrements d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH** ou de la commande PCF **Set Channel Authentication Record**.

Remarque : Un grand nombre d'enregistrements d'authentification de canal peut avoir un impact négatif sur les performances d'un gestionnaire de files d'attente.

Blocage d'adresses IP

C'est normalement le rôle d'un pare-feu que de prévenir l'accès provenant de certaines adresses IP. Toutefois, il peut arriver que vous constatiez des tentatives de connexion provenant d'une adresse IP qui ne devrait pas avoir accès au système IBM MQ et que vous deviez temporairement bloquer l'adresse avant que le pare-feu ne puisse être mis à jour. Ces tentatives de connexion peuvent ne pas provenir de canaux IBM MQ, mais d'autres applications socket mal configurées pour cibler votre programme d'écoute IBM MQ. Bloquez les adresses IP en définissant un enregistrement d'authentification de canal de type BLOCKADDR. Vous pouvez spécifier une ou plusieurs adresses, ou des modèles avec des caractères génériques.

Lorsqu'une connexion entrante est refusée en raison d'un blocage de l'adresse IP de cette manière, un message d'événement MQRC_CHANNEL_BLOCKED avec un qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_ADDRESS généré, à condition que les événements du canal soient activés et que le gestionnaire de files d'attente soit en cours d'exécution. En outre, la connexion reste ouverte pendant 30 secondes avant de renvoyer l'erreur afin de garantir que le programme d'écoute n'est pas saturé par les tentatives de connexion répétées qui sont bloquées.

Pour bloquer des adresses IP uniquement sur des canaux spécifiques ou pour éviter le délai avant le signalement de l'erreur, définissez un enregistrement d'authentification de canal de type ADDRESSMAP avec le paramètre USERSRC(NOACCESS).

Lorsqu'une connexion entrante est refusée pour cette raison, un message d'événement MQRC_CHANNEL_BLOCKED avec le qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_NOACCESS est généré, à condition que les événements du canal soient activés et que le gestionnaire de files d'attente soit en cours d'exécution.

Pour voir un exemple, consultez «Blocage d'adresses IP spécifiques», à la page 397.

Blocage d'ID utilisateur

Pour empêcher certains ID utilisateur de se connecter sur un canal client, définissez un enregistrement d'authentification du canal de type BLOCKUSER. Ce type d'enregistrement s'applique uniquement aux canaux client disponibles et non aux canaux de message. Vous avez la possibilité d'indiquer un ou plusieurs ID utilisateur individuels, mais pas d'utiliser de caractères génériques.

Chaque fois qu'une connexion entrante est refusée pour cette raison, un message d'événement MQRC_CHANNEL_BLOCKED avec un qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_USERID est généré, à condition que les événements du canal soient activés.

Pour voir un exemple, consultez «Blocage d'ID utilisateur spécifiques», à la page 399.

Vous pouvez également bloquer l'accès d'un ID utilisateur quelconque sur certains canaux en définissant un enregistrement d'authentification du canal de type USERMAP avec le paramètre USERSRC(NOACCESS).

Lorsqu'une connexion entrante est refusée pour cette raison, un message d'événement MQRC_CHANNEL_BLOCKED avec le qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_NOACCESS est généré, à condition que les événements du canal soient activés et que le gestionnaire de files d'attente soit en cours d'exécution.

Pour voir un exemple, consultez «Blocage de l'accès pour un ID utilisateur client», à la page 402.

Blocage de gestionnaires de files d'attente

Pour bloquer l'accès à un canal se connectant à partir d'un gestionnaire de files d'attente spécifique, définissez un enregistrement d'authentification de canal de type QMGRMAP avec le paramètre USERSRC(NOACCESS). Vous pouvez indiquer un nom de gestionnaire de files d'attente ou un modèle comportant des caractères génériques. La fonction BLOCKUSER permettant de bloquer l'accès d'un gestionnaire de files d'attente ne comporte pas d'équivalent.

Lorsqu'une connexion entrante est refusée pour cette raison, un message d'événement MQRC_CHANNEL_BLOCKED avec le qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_NOACCESS est généré, à condition que les événements du canal soient activés et que le gestionnaire de files d'attente soit en cours d'exécution.

Pour voir un exemple, consultez «Blocage de l'accès à partir d'un gestionnaire de files d'attente éloignées», à la page 401.

Blocage des noms distinctifs SSL ou TLS

Pour bloquer l'accès à un utilisateur présentant un certificat personnel SSL ou TLS doté d'un nom distinctif spécifique, définissez un enregistrement d'authentification de canal de type SSLPEERMAP avec le paramètre USERSRC(NOACCESS). Vous pouvez indiquer un nom distinctif unique ou un modèle comportant des caractères génériques. La fonction BLOCKUSER permettant de bloquer l'accès aux noms distinctifs ne comporte pas d'équivalent.

Lorsqu'une connexion entrante est refusée pour cette raison, un message d'événement MQRC_CHANNEL_BLOCKED avec le qualificateur de code anomalie MQRQ_CHANNEL_BLOCKED_NOACCESS est généré, à condition que les événements du canal soient activés et que le gestionnaire de files d'attente soit en cours d'exécution.

Pour voir un exemple, consultez «Blocage de l'accès pour un nom distinctif SSL ou TLS», à la page 403.

Mappage d'adresses IP vers les ID utilisateur requis

Pour indiquer qu'un canal se connectant à partir d'une adresse IP spécifique doit utiliser une valeur MCAUSER spécifique, définissez un enregistrement d'authentification de canal de type ADDRESSMAP. Vous pouvez indiquer une adresse unique, une plage d'adresses ou un modèle comportant des caractères génériques.

Si vous utilisez un réexpéditeur de port, une rupture de session DMZ ou toute autre configuration modifiant l'adresse IP présentée au gestionnaire de files d'attente, le mappage des adresses IP ne convient pas forcément à votre situation.

Pour voir un exemple, consultez «Mappage d'une adresse IP à un ID utilisateur MCAUSER», à la page 403.

Mappage de gestionnaires de files d'attente vers les ID utilisateur requis

Pour indiquer qu'un canal se connectant à partir d'un gestionnaire de files d'attente spécifique doit utiliser une valeur MCAUSER spécifique, définissez un enregistrement d'authentification de canal de type QMGRMAP. Vous pouvez indiquer un nom de gestionnaire de files d'attente ou un modèle comportant des caractères génériques.

Pour voir un exemple, consultez «Mappage d'un gestionnaire de files d'attente éloignées à un ID utilisateur MCAUSER», à la page 399.

Mappage des ID utilisateur vérifiés par un client vers les ID utilisateur requis

Pour indiquer qu'un ID utilisateur se connectant à partir d'un client IBM MQ MQI doit utiliser une valeur MCAUSER différente, définissez un enregistrement d'authentification de canal de type USERMAPP. Le mappage d'ID utilisateur ne se sert pas de caractères génériques.

Pour voir un exemple, consultez «Mappage d'un ID utilisateur client à un ID utilisateur MCAUSER», à la page 400.

Mappage des noms distinctifs SSL ou TLS vers les ID utilisateur requis

Pour indiquer qu'un utilisateur présentant un certificat personnel SSL/TLS doté d'un nom distinctif doit utiliser une valeur MCAUSER spécifique, définissez un enregistrement d'authentification de canal de type SSLPEERMAP. Vous pouvez indiquer un nom distinctif unique ou un modèle comportant des caractères génériques.

Pour voir un exemple, consultez «Mappage d'un nom distinctif SSL ou TLS à un ID utilisateur MCAUSER», à la page 401.

Mappage de gestionnaires de files d'attente, de clients ou de noms distinctifs SSL ou TLS en fonction d'une adresse IP

Dans certains, il est possible qu'une tierce partie falsifie le nom d'un gestionnaire de files d'attente. Un certificat SSL ou TLS ou un fichier de clés peut également être volé et réutilisé. Pour vous protéger contre ces menaces, vous pouvez indiquer qu'une connexion provenant d'un certain gestionnaire de files d'attente ou client ou utilisant un certain nom distinctif doit être établie à partir d'une adresse IP spécifique. Définissez un enregistrement d'authentification de canal de type USERMAP, QMGRMAP ou SSLPEERMAP et spécifiez l'adresse IP ou le modèle d'adresse autorisé à l'aide du paramètre ADDRESS.

Pour voir un exemple, consultez «Mappage d'un gestionnaire de files d'attente éloignées à un ID utilisateur MCAUSER», à la page 399.

Interaction entre les enregistrements d'authentification de canal

Il se peut qu'un canal tentant de se connecter corresponde à plusieurs enregistrements d'authentification de canaux et que leurs effets soient contradictoires. Par exemple, un canal peut vérifier un ID utilisateur qui soit bloqué par un enregistrement BLOCKUSER, mais qui comporte un certificat SSL ou TLS correspondant à un enregistrement SSLPEERMAP qui définisse un ID utilisateur différent. De plus, si les enregistrements utilisent des caractères génériques, il se peut qu'une adresse IP unique, un gestionnaire de files d'attente ou un nom distinctif SSL ou TLS corresponde à plusieurs modèles. Par exemple, l'adresse IP 192.0.2.6 correspond aux modèles 192.0.2.0-24, 192.0.2. *, et 192.0. * 6. L'action prise varie en fonction des éléments ci-dessous.

- L'enregistrement d'authentification de canal est sélectionné de la manière suivante :
  - Un enregistrement d'authentification de canal correspondant de manière explicite au nom du canal est prioritaire sur un enregistrement dont le canal comporte des caractères génériques.
  - Un enregistrement d'authentification de canal portant un nom distinctif SSL ou TLS est prioritaire sur un enregistrement associé à un ID utilisateur, un gestionnaire de files d'attente ou une adresse IP.
  - Un enregistrement d'authentification de canal associé à un ID utilisateur ou un gestionnaire de files d'attente est prioritaire sur un enregistrement faisant appel à une adresse IP.
- Si un enregistrement d'authentification de canal équivalent est détecté et qu'il indique une valeur MCAUSER, cette dernière est affectée au canal.
- Si un enregistrement d'authentification de canal équivalent est détecté et qu'il indique que le canal ne dispose d'aucun droit d'accès, une valeur MCAUSER de type *NOACCESS est affectée au canal. Cette valeur peut ensuite être modifiée par un programme exit de sécurité.
- Si aucun enregistrement d'authentification équivalent n'est détecté, ou si un enregistrement équivalent est détecté et qu'il indique que l'ID utilisateur du canal doit être utilisé, la zone MCAUSER est examinée.
  - Si la zone MCAUSER est vide, l'ID utilisateur client est affecté au canal.
  - Si la zone MCAUSER n'est pas vide, sa valeur est affectée au canal.

- Un programme exit de sécurité est exécuté. Ce programme peut définir l'ID utilisateur du canal ou déterminer si l'accès doit être bloqué.
- Si la connexion est bloquée ou si la valeur MCAUSER est définie sur *NOACCESS, le canal s'arrête.
- Si la connexion n'est pas bloquée, l'ID utilisateur du canal défini à l'étape précédente est vérifié par rapport à la liste des utilisateurs bloqués et ce, pour tous les canaux à l'exception d'un canal client.
  - Si l'ID utilisateur figure dans la liste des utilisateurs bloqués, le canal s'arrête.
  - Si l'ID utilisateur ne figure pas dans la liste des utilisateurs bloqués, le canal s'exécute.

Lorsque plusieurs enregistrements d'authentification de canal correspondent à un nom de canal, une adresse IP, un nom d'hôte, un nom de gestionnaire de files d'attente ou un nom distinctif SSL ou TLS, la correspondance la plus spécifique est utilisée. La correspondance considérée comme :

- La plus spécifique est un nom ne comportant pas de caractère générique, par exemple :
  - Un nom de canal A.B.C
  - Une adresse IP 192.0.2.6
  - Un nom d'hôte `hursley.ibm.com`
  - Un nom de gestionnaire de files d'attente 192.0.2.6
- La plus générique est l'astérisque unique (*), qui correspond, par exemple, à :
  - Tous les noms de canal
  - Toutes les adresses IP
  - Tous les noms d'hôte
  - Tous les noms de gestionnaire de files d'attente
- Un pattern de type chaîne commençant par un astérisque est plus générique qu'une chaîne dont le début est une valeur définie :
  - Pour les canaux, *.B.C est plus générique que A.*
  - Pour les adresses IP, *.0.2.6 est plus générique que 192.*
  - Pour les noms d'hôte, `*.ibm.com` est plus générique que `hursley.*`
  - Pour les noms de gestionnaire de files d'attente, *QUEUEMANAGER est plus générique que QUEUEMANAGER*
- Un pattern comportant un astérisque à une position spécifique dans une chaîne est plus générique qu'un pattern comportant une valeur définie à la même position dans une chaîne ; il en va de même pour chaque position suivante dans une chaîne :
  - Pour les canaux, A.*.C est plus générique que A.B.*
  - Pour les adresses IP, 192.*.2.6 est plus générique que 192.0.*.
  - Pour les noms d'hôte, `hursley.*.com` est plus générique que `hursley.ibm.*`
  - Pour les noms de gestionnaire de files d'attente, Q*MANAGER est plus générique que QUEUE*
- Lorsque plusieurs patterns comportent un astérisque à une position spécifique dans une chaîne, le pattern qui comporte le nombre de noeuds le moins élevé après l'astérisque est le plus générique :
  - Pour les canaux, A.* est plus générique que A.*.C
  - Pour les adresses IP, 192.* est plus générique que 192.*.2.*.
  - Pour les noms d'hôte, `hurlsey.*` est plus générique que `hursley.*.com`
  - Pour les noms de gestionnaire de files d'attente, Q* est plus générique que Q*MGR
- De plus, pour une adresse IP :
  - Une plage signalée par un trait d'union (-) est plus spécifique qu'un astérisque. Ainsi, 192.0.2.0-24 est plus spécifique que 192.0.2.*.
  - Une plage représentant un sous-ensemble d'une autre plage est plus spécifique que la plage la plus grande. Ainsi, 192.0.2.5-15 est plus spécifique que 192.0.2.0-24.

- Les plages qui se chevauchent ne sont pas autorisées. Par exemple, vous ne pouvez pas avoir d'enregistrements d'authentification de canaux pour 192.0.2.0-15 et 192.0.2.10-20.
- Un modèle ne peut pas contenir moins d'éléments que ce qui est obligatoire, sauf si le modèle se termine par une astérisque. Par exemple 192.0.2 n'est pas valide, mais 192.0.2.* est valide.
- Un astérisque de fin doit être séparé du reste de l'adresse par le séparateur d'élément approprié (un point (.) pour IPv4, un deux-points (:) pour IPv6). Par exemple, 192.0* n'est pas valide parce que l'astérisque n'est pas un élément en soi.
- Un pattern peut contenir des astérisques supplémentaires, à condition qu'aucun ne soit adjacent à l'astérisque de fin. Par exemple, 192.*.2.* est valide, mais 192.0.*.* est incorrect.
- Un pattern d'adresse IPv6 ne peut pas contenir le signe deux-points et un astérisque de fin, car l'adresse résultante serait ambiguë. Par exemple, 2001::* pourrait devenir 2001:0000:*, 2001:0000:0000:* etc.

- Pour un nom distinctif SSL ou TLS, l'ordre de priorité des sous-chaînes est le suivant :

Tableau 7. Ordre de priorité des sous-chaînes

Commande	**Sous-chaîne de nom distinctif**	**Nom**
1	SERIALNUMBER=	Numéro de série du certificat
2	MAIL=	Adresse électronique
3	E=	Adresse électronique (dépréciée dans la préférence pour MAIL)
4	UID=, USERID=	ID utilisateur
5	CN=	Nom usuel
6	T =	Titre
7	OU=	Unité organisationnelle
8	DC=	Composant de domaine
9	O=	Organisation
10	STREET=	Rue/Première ligne d'adresse
11	L=	Localité
12	ST=, SP=, S=	Nom de département
13	PC =	Code postal
14	C=	Pays
15	UNSTRUCTUREDNAME=	Nom d'hôte
16	UNSTRUCTUREDADDRESS=	Adresse IP
17	DNQ=	Qualificateur de nom distinctif

Par conséquent, si un certificat SSL ou TLS se présente avec un nom distinctif comportant les sous-chaînes O=IBM et C=UK, IBM MQ utilise un enregistrement d'authentification de canal pour O=IBM, mais pas pour C=UK si les deux sont présents.

Un nom distinctif peut contenir plusieurs OU, qui doit être spécifiée dans un ordre hiérarchique, les unités organisationnelles les plus grandes spécifiées en premier. Si deux noms distinctifs sont équivalents en tous points sauf en ce qui concerne leurs valeurs d'unités organisationnelles, le nom distinctif le plus spécifique est déterminé comme suit :

1. S'ils ont des nombres d'attributs d'unités organisationnelles différents, le nom distinctif possédant le plus de valeurs d'unités organisationnelles est le plus spécifique. Cela vient du fait que le nom distinctif possédant le plus d'unités organisationnelles qualifie le nom distinctif plus en détails et

apporte plus de critères de correspondance. Même si l'unité organisationnelle de niveau supérieure est un caractère générique (OU=*), le nom distinctif possédant le plus d'unités organisationnelles est toujours considéré comme le plus spécifique globalement.

2. S'ils ont le même nombre d'attributs d'unités organisationnelles, les paires de valeurs d'unités organisationnelles correspondantes sont comparées dans l'ordre de gauche à droite, celles de gauche étant celles de plus haut niveau (les moins spécifiques), selon les règles suivantes.
   a. Une unité organisationnelle sans valeur indiquée par des caractères génériques est la plus spécifique car elle ne peut correspondre qu'à une seule chaîne exacte.
   b. Une unité organisationnelle avec un seul caractère générique, que ce soit au début ou à la fin (par exemple OU=ABC* ou OU=*ABC) est la plus spécifique suivante.
   c. Une unité organisationnelle avec deux caractères génériques par exemple OU=*ABC*) est la plus spécifique qui suit.
   d. Une unité organisationnelle constituée d'un seul astérisque (OU=*) est la moins spécifique.
3. Si la comparaison de chaîne est liée entre deux valeurs d'attributs de la même spécificité, alors la chaîne d'attribut la plus longue sera la plus spécifique.
4. Si la comparaison de chaîne est liée entre deux valeurs d'attributs de la même spécificité et de même longueur, le résultat est déterminé par une comparaison de chaîne ne respectant pas la casse de la portion de nom distinctif d'où sont exclus les caractères génériques.

Si deux DN sont égaux à tous égards, à l'exception de leurs valeurs de DC, les mêmes règles de correspondance s'appliquent que pour les OU, sauf que dans les valeurs de DC, la DC de gauche est le niveau le plus bas (le plus spécifique) et l'ordre de comparaison diffère en conséquence.

Affichage des enregistrements d'authentification de canaux

Pour afficher les enregistrements d'authentification de canaux, utilisez la commande MQSC **DISPLAY CHLAUTH** ou la commande PCF **Inquire Channel Authentication Records**. Vous pouvez choisir de renvoyer tous les enregistrements qui correspondent au nom de canal fourni ou seulement ceux qui correspondent à un élément particulier. La correspondance explicite vous indique quel enregistrement d'authentification de canal est utilisé lorsqu'un canal tente d'établir une connexion à partir d'une adresse IP ou d'un gestionnaire de files d'attente spécifique, qu'il utilise un ID utilisateur spécifique et qu'il présente un certificat personnel SSL/TLS doté d'un nom distinctif, le cas échéant.

Concepts associés

«Sécurité de la messagerie distante», à la page 97
Cette section traite des aspects de la sécurité liés à la messagerie distante.

Interaction entre CHLAUTH et CONNAUTH

Comment les enregistrements d'authentification de canal (CHLAUTH) et l'authentification de connexion (CONNAUTH) interagissent dans IBM MQ, dans le cas d'une conversation unique sur un canal.

Différents types de liaisons

IBM MQ prend en charge deux méthodes permettant à une application de se connecter:

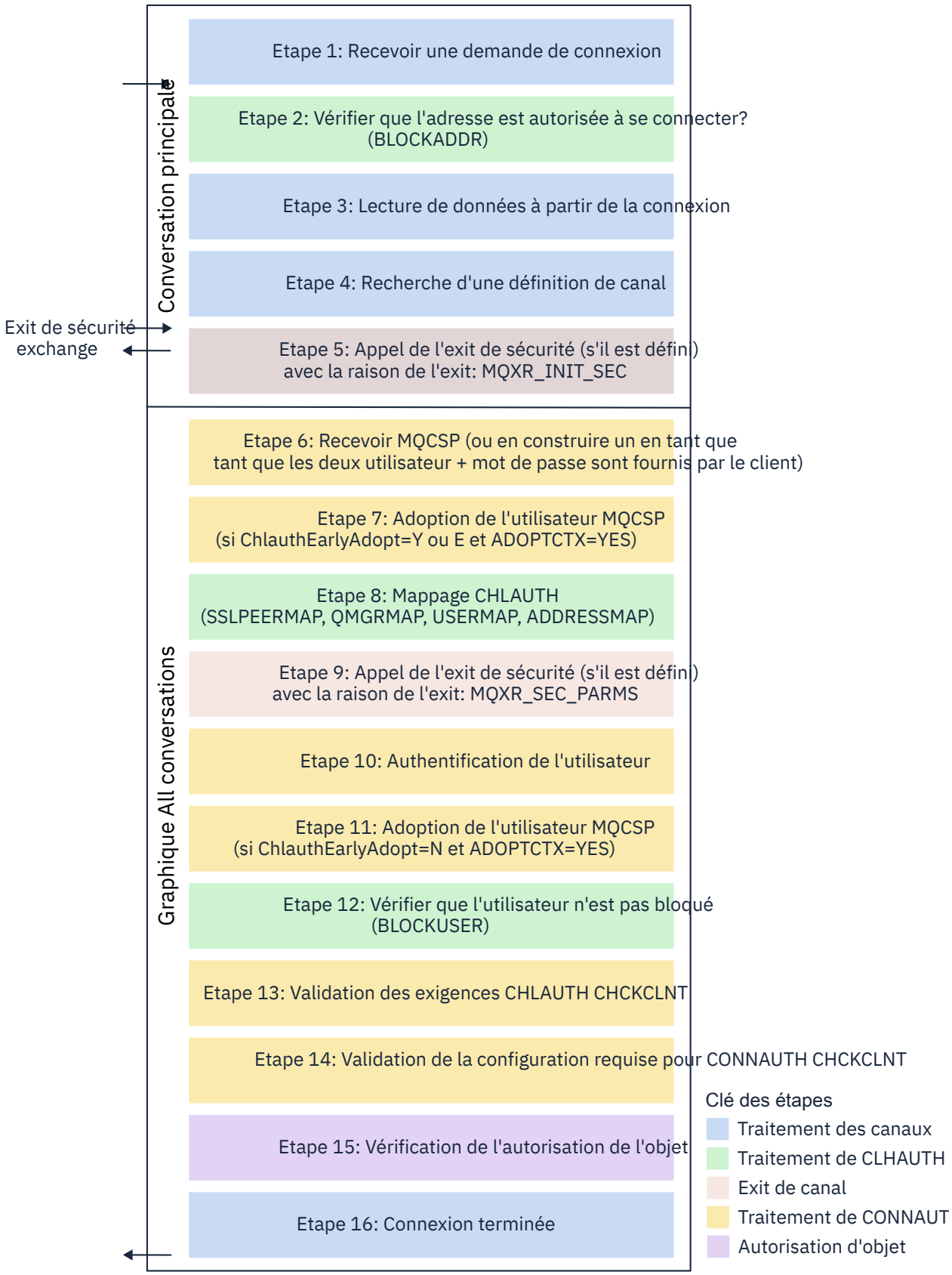
Liaisons locales
S'applique lorsque l'application et le gestionnaire de files d'attente se trouvent sur la même image d'exploitation. CHLAUTH n'est pas pertinent pour ce type de connexion d'application.

Liaisons client
S'applique lorsque l'application et le gestionnaire de files d'attente utilisent le réseau pour communiquer. L'application et le gestionnaire de files d'attente peuvent s'exécuter sur la même machine ou sur des machines différentes. Dans IBM MQ, une connexion client est gérée sous la forme d'un canal de connexion serveur (SVRCONN) et, dans ce cas, CONNAUTH et CHLAUTH sont applicables.

Etapes de liaison de l'extrémité réceptrice d'un canal

Lorsqu'une application se connecte à un gestionnaire de files d'attente, une vérification importante est effectuée pour s'assurer que les deux extrémités du canal comprennent ce qui est pris en charge par l'autre extrémité. L'extrémité réceptrice du canal effectue une vérification supplémentaire, impliquant CHLAUTH et CONNAUTH, pour s'assurer que le client est autorisé à se connecter, et ce processus peut également inclure un exit de sécurité car cela peut affecter le résultat. Cette phase de connexion de canal est également appelée *phase de liaison*.

Le diagramme suivant répertorie les étapes par lesquelles passe un canal SVRCONN lorsque l'extrémité du serveur (au niveau du gestionnaire de files d'attente) démarre:



Etape 1: Recevoir une demande de connexion
L'initiateur de canal ou le programme d'écoute reçoit une demande de connexion de quelque part sur le réseau.

Etape 2: L'adresse est-elle autorisée à se connecter?
Avant toute lecture de données, IBM MQ vérifie l'adresse IP du partenaire par rapport aux règles CHLAUTH, pour voir si l'adresse se trouve dans la règle *BLOCKADDR* . Si l'adresse est introuvable et donc non bloquée, le flux passe à l'étape suivante.

Etape 3: Lecture de données à partir du canal
IBM MQ lit maintenant les données dans une mémoire tampon et commence à traiter les informations envoyées.

Etape 4: Recherche de la définition de canal
Dans le premier flux de données, IBM MQ envoie, entre autres, le nom du canal que la fin émettrice tente de démarrer. Le gestionnaire de files d'attente de réception peut ensuite rechercher la définition de canal, qui possède tous les paramètres spécifiés pour le canal.

Etape 5: Appel de l'exit de sécurité (s'il est défini)
Si un exit de sécurité (SCYEXIT) est défini pour le canal, il est appelé avec la raison de l'exit (MQCXP.**ExitReason**) défini sur MQXR_INIT_SEC.

Etape 6: Réception de MQCSP
Si nécessaire, construisez un, tant que l'ID utilisateur et le mot de passe sont fournis par le client.

Si le client est une application Java ou JMS s'exécutant en mode compatibilité, il ne transmet pas de structure MQCSP au gestionnaire de files d'attente. A la place, si l'application a fourni un ID utilisateur et un mot de passe, une structure MQCSP est construite ici.

Etape 7: Adoptez l'utilisateur MQCSP (si `ChlauthEarlyAdopt` a pour valeur *Y* et ADOPTCTX=YES)
L'ID utilisateur vérifié par le client est authentifié.

Si CONNAUTH utilise LDAP pour mapper un nom distinctif vérifié à un ID utilisateur court, le mappage est effectué dans cette étape.

Si l'authentification aboutit, l'ID utilisateur est adopté par le canal et utilisé par l'étape de mappage CHLAUTH.

Remarque : A partir de IBM MQ 9.0.4 , le paramètre **`ChlauthEarlyAdopt=`** *Y* est automatiquement ajouté à la strophe channels du fichier qm.ini pour les nouveaux gestionnaires de files d'attente.

Etape 8: Mappage CHLAUTH
Le cache CHLAUTH est à nouveau inspecté pour rechercher les règles de mappage *SSLPEERMAP*, *USERMAP*, *QMGRMAP*et *ADDRESSMAP*.

La règle qui correspond le plus spécifiquement au canal entrant est utilisée. Si la règle comporte **USERSRC***(CHANNEL)* ou *(MAP)*, le canal se poursuit lors de la liaison.

Si les règles CHLAUTH ont pour résultat une règle avec **USERSRC***(NOACCESS)*, l'application ne peut pas se connecter au canal, sauf si les données d'identification sont remplacées par un ID utilisateur et un mot de passe valides à l'étape 9.

Etape 9: Appel de l'exit de sécurité (s'il est défini)
Si un exit de sécurité (SCYEXIT) est défini pour le canal, il est appelé avec la raison de l'exit (MQCXP.**ExitReason**) défini sur MQXR_SEC_PARMS.

Un pointeur vers MQCSP sera présent dans la zone **SecurityParms** de la structure MQCXP.

La structure MQCSP comporte des pointeurs vers l'ID utilisateur (MQCSP.**CSPUserIdPtr**) et le mot de passe (MQCSP.**CSPPasswordPtr**).

Il est possible de modifier l'ID utilisateur et le mot de passe dans l'exit. L'exemple suivant montre comment un exit de sécurité imprime les valeurs d'ID utilisateur et de mot de passe dans un journal d'audit:

```
if (pMQCXP -> ExitReason == MQXR_SEC_PARMS)
{
  /* It is not a good idea for security reasons to print out the user ID */
  /* and password but the following is shown for demonstration reasons */
```

```
  printf("User ID: %.*s Password: %.*s\n",
         pMQCXP -> SecurityParms -> CSPUserIdLength,
         pMQCXP -> SecurityParms -> CSPUserIdPtr,
         pMQCXP -> SecurityParms -> CSPPasswordLength,
         pMQCXP -> SecurityParms -> CSPPasswordPtr);
```

L'exit peut demander à IBM MQ de fermer le canal en renvoyant *MQXCC_CLOSE_CHANNEL* dans MQCXP.Zone **Exitresponse** . Sinon, le traitement du canal se poursuit jusqu'à la phase d'authentification de la connexion.

Remarque : Si l'utilisateur vérifié est modifié par l'exit de sécurité, les règles de mappage CHLAUTH ne seront pas réappliquées au nouvel utilisateur.

Etape 10: Authentification de l'utilisateur

La phase d'authentification se produit si CONNAUTH est activé sur le gestionnaire de files d'attente.

Pour vérifier cela, exécutez la commande MQSC'DISPLAY QMGR CONNAUTH'.

z/OS L'exemple suivant illustre la sortie de la commande **DISPLAY QMGR CONNAUTH** à partir d'un gestionnaire de files d'attente s'exécutant sous IBM MQ for z/OS.

```
 CSQM201I !MQ25 CSQMDRTC DISPLAY QMGR DETAILS
 QMNAME(MQ25)
 CONNAUTH(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
  END QMGR DETAILS
 CSQ9022I !MQ25 CSQMDRTC ' DISPLAY QMGR' NORMAL COMPLETION
```

Multi L'exemple suivant illustre la sortie de la commande'**DISPLAY QMGR CONNAUTH**'à partir d'un gestionnaire de files d'attente s'exécutant sous IBM MQ for Multiplatforms.

```
     1 : DISPLAY QMGR CONNAUTH
AMQ8408: Display Queue Manager details.
   QMNAME(DEMO)
   CONNAUTH(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
```

La valeur CONNAUTH est le nom d'un objet **AUTHINFO** IBM MQ .

Comme l'authentification du système d'exploitation (**AUTHTYPE**(*IDPWOS*)) est valide sur IBM MQ for Multiplatforms et IBM MQ for z/OS, les exemples utilisent l'authentification du système d'exploitation.

z/OS L'exemple suivant illustre l'objet par défaut fourni pour **AUTHTYPE**(*IDPWOS*) à partir d'un gestionnaire de files d'attente s'exécutant sous IBM MQ for z/OS.

```
 CSQM293I !MQ25 CSQMDRTC 1 AUTHINFO FOUND MATCHING REQUEST CRITERIA
 CSQM201I !MQ25 CSQMDRTC DISPLAY AUTHINFO DETAILS
 AUTHINFO(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
 AUTHTYPE(IDPWOS)
 QSGDISP(QMGR)
 ADOPTCTX(NO)
 CHCKCLNT(NONE)
 CHCKLOCL(OPTIONAL)
 FAILDLAY(1)
 DESCR()
 ALTDATE(2018-06-04)
 ALTTIME(10.43.04)
   END AUTHINFO DETAILS
 CSQ9022I !MQ25 CSQMDRTC ' DISPLAY AUTHINFO' NORMAL COMPLETION
```

Multi L'exemple suivant illustre l'objet par défaut fourni pour **AUTHTYPE**(*IDPWOS*) à partir d'un gestionnaire de files d'attente s'exécutant sous IBM MQ for Multiplatforms.

```
 1 : display authinfo(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
AMQ8566: Display authentication information details.
   AUTHINFO(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
   AUTHTYPE(IDPWOS)                        ADOPTCTX(NO)
   DESCR( )                                CHCKCLNT(REQDADM)
```

```
CHCKLOCL(OPTIONAL)                  FAILDLAY(1)
ALTDATE(2015-06-08)                 ALTTIME(16.35.16)
```

L'élément AUTHINFO TYPE (IDPWOS) possède un attribut appelé CHCKCLNT. Si la valeur est remplacée par *REQUIRED* , toutes les applications client doivent fournir un ID utilisateur et un mot de passe valides.

Si l'utilisateur a été authentifié à l'étape 7, il ne sera pas à nouveau authentifié sauf si l'utilisateur ou le mot de passe de la zone SecurityParms de la structure MQCXP a été modifié par un exit de sécurité à l'étape 9.

Etape 11: Adopter le contexte de l'utilisateur MQCSP (si `ChlauthEarlyAdopt=`*N* et ADOPTCTX=YES)
Vous pouvez définir l'attribut ADOPTCTX , qui contrôle si le canal s'exécute sous MCAUSER ou l'ID utilisateur fourni par l'application.

Si l'ID utilisateur vérifié dans la zone MQCSP ou **`SecurityParms`** de la structure MQXCP a été authentifié avec succès et que **ADOPTCTX** est défini sur *YES*, le contexte de l'utilisateur résultant des étapes 7 et 8 est adopté comme contexte à utiliser pour cette application, sauf si l'utilisateur ou le mot de passe de la zone **`SecurityParms`** de la structure MQCXP a été modifié par un exit de sécurité à l'étape 9.

Cet ID utilisateur vérifié correspond à l'ID utilisateur vérifié pour l'autorisation d'utiliser les ressources IBM MQ .

Par exemple, vous n'avez pas de MCAUSER défini sur le canal SVRCONN et votre client s'exécute sous`johndoe`sur votre machine Linux . Votre application spécifie l'utilisateur`fred`dans le MQCSP, de sorte que le canal commence à s'exécuter avec`johndoe`en tant que MCAUSER actif. Après la vérification CONNAUTH, l'utilisateur`fred`est adopté et le canal s'exécute avec`fred`comme utilisateur MCAUSER actif.

Etape 12: Vérifier que l'utilisateur n'est pas bloqué (BLOCKUSER)
Si la vérification **CONNAUTH** aboutit, le cache CHLAUTH est à nouveau inspecté pour vérifier si le MCAUSER actif est bloqué par une règle *BLOCKUSER* . Si l'utilisateur est bloqué, le canal se termine.

Step13: Validation des exigences CHLAUTH CHCKCLNT
Si la règle CHLAUTH sélectionnée à l'étape 8 spécifie en plus une valeur CHCKCLNT de REQUIRED ou REQDADM, la validation est effectuée pour s'assurer qu'un ID utilisateur CONNAUTH valide a été fourni pour répondre à l'exigence.

- Si CHCKCLNT (REQUIRED) est défini, un utilisateur doit avoir été authentifié à l'étape 7 ou 10. Sinon, la connexion est rejetée.
- Si CHCKCLNT (REQDADM) est défini, un utilisateur doit avoir été authentifié à l'étape 7 ou 10 si cette connexion est privilégiée. Sinon, la connexion est rejetée.
- Si CHCKCLNT (ASQMGR) est défini, cette étape est ignorée.

Remarques :

1. Si CHCKCLNT (REQUIRED) ou CHCKCLNT (REQDADM) est défini, mais que CONNAUTH n'est pas activé sur le gestionnaire de files d'attente, la connexion échoue avec un code retour MQRC_SECURITY_ERROR (2063) en raison du conflit dans la configuration.
2. L'utilisateur n'est pas réauthentifié dans cette étape.

Etape 14: Validez les exigences de CONNAUTH CHCKCLNT.
La phase d'authentification se produit si CONNAUTH est activé sur le gestionnaire de files d'attente.

La valeur de CONNAUTH CHCKCLNT est vérifiée pour déterminer les exigences définies pour les connexions entrantes:

- Si CHCKCLNT (NONE) est défini, cette étape est ignorée.
- Si CHCKCLNT (OPTIONAL) est défini, cette étape est ignorée.
- Si CHCKCLNT (REQUIRED) est défini, un utilisateur doit avoir été authentifié à l'étape 7 ou 10. Sinon, la connexion est rejetée.
- Si CHCKCLNT (REQDADM) est défini, un utilisateur doit avoir été authentifié à l'étape 7 ou 10 si cette connexion est privilégiée. Sinon, la connexion est rejetée.

Remarque : L'utilisateur n'est pas réauthentifié dans cette étape.

Multi **Etape 15: Vérifier l'autorisation de l'objet**
Une vérification est effectuée pour s'assurer que le MCAUSER actif dispose des droits appropriés pour se connecter au gestionnaire de files d'attente.

ULW Pour plus d'informations, voir Gestionnaire des droits d'accès aux objets.

IBM i Pour plus d'informations, voir «Gestionnaire des droits d'accès aux objets sous IBM i», à la page 158.

Etape 16: La connexion est terminée
Si les étapes précédentes aboutissent, la connexion se termine.

Concepts associés
CONNAUTH
Un gestionnaire de files d'attente peut être configuré pour utiliser un ID utilisateur et un mot de passe fournis afin de vérifier si un utilisateur dispose des droits d'accès aux ressources.

Référence associée
SET CHLAUTH
ALTER AUTHINFO

Résolution des problèmes d'accès CHLAUTH
Suggestions sur la façon de résoudre certains problèmes d'accès lors de l'utilisation d'enregistrements d'authentification de canal (CHLAUTH).

Règles CHLAUTH par défaut

Il existe trois règles par défaut pour le traitement CHLAUTH:

- AUCUN ACCES à tous les canaux par les utilisateurs `MQ-admin*`
- AUCUN ACCES à tous les SYSTEM.* canaux par tous les utilisateurs
- Accès ALLOW à SYSTEM.ADMIN.SVRCONN (non utilisateurs `MQ-admin`)

Les deux premières règles bloquent l'accès à tous les canaux. La troisième règle est plus spécifique et est donc prioritaire sur les deux autres, si le canal est SYSTEM.ADMIN.SVRCONN ADMIN.SVRCONN, permettant ainsi l'accès à ce canal.

Erreurs de connexion courantes

Les règles HLAUTH permettent de déterminer si un canal peut être démarré et d'autoriser le mappage via MCAUSER vers un autre ID utilisateur. Si le canal ne peut pas être démarré, les erreurs suivantes se produisent généralement:

- RC 2035 MQRC_NOT_AUTHORIZED
- RC 2059 MQRC_Q_MGR_NOT_AVAILABLE
- AMQ4036 Accès non autorisé
- AMQ9776: Le canal a été bloqué par l'ID utilisateur
- AMQ9777: Le canal a été bloqué
- MQJE001: Une exception MQException s'est produite: Code achèvement 2, Motif 2035
- MQJE036: Le gestionnaire de files d'attente a rejeté la tentative de connexion

Vous devez bloquer l'accès strictement, puis ajouter d'autres règles CHLAUTH pour contrôler qui peut accéder aux canaux et les démarrer. A titre de mesure temporaire, et pour identifier et résoudre les erreurs répertoriées, vous pouvez:

- «Désactiver les règles CHLAUTH», à la page 63
- «Modifier ou supprimer des règles CHLAUTH», à la page 63

Désactiver les règles CHLAUTH

A titre de mesure temporaire, ainsi que pour identifier et résoudre les erreurs ci-dessus, vous pouvez désactiver les règles CHLAUTH. Les règles peuvent être réactivées à tout moment et si la désactivation des règles CHLAUTH résout le problème de connexion, vous savez que c'est la cause.

Pour désactiver les règles CHLAUTH, exécutez la commande suivante:

```
runmqsc: ALTER QMGR CHLAUTH (DISABLED)
```

Notez que vous pouvez également définir CHLAUTH sur *WARN*, qui autorise l'accès et consigne le résultat de la règle.

Modifier ou supprimer des règles CHLAUTH

Vous pouvez également supprimer ou modifier la ou les règles CHLAUTH à l'origine de votre problème.

Pour modifier une règle CHLAUTH, utilisez la commande SET CHLAUTH avec ACTION (REPLACE). Par exemple, pour modifier la règle par défaut qui empêche tous les utilisateurs `MQ-admin` d'accéder à tous les canaux vers WARN, au lieu d'être bloquée, exécutez la commande suivante:

```
runmqsc: SET CHLAUTH (*) TYPE (BLOCKUSER) USERLIST (*MQADMIN) WARN(YES)
ACTION (REPLACE)
```

Pour supprimer une règle CHLAUTH, utilisez la commande SET CHLAUTH avec ACTION (REMOVE). Par exemple, pour supprimer la règle par défaut qui empêche tous les utilisateurs `MQ-admin` d'accéder à tous les canaux, exécutez la commande suivante:

```
runmqsc: SET CHLAUTH (*) TYPE (BLOCKUSER) USERLIST (*MQADMIN) ACTION (REMOVE)
```

Test de l'accès à l'aide de MATCH (RUNCHECK)

Vous pouvez tester le résultat de vos règles CHLAUTH à l'aide de l'option MATCH (*RUNCHECK*) de la règle CHLAUTH dans runmqsc. L'option **MATCH** (*RUNCHECK*) renvoie l'enregistrement correspondant à un canal entrant spécifique lors de l'exécution, si ce canal se connecte à ce gestionnaire de files d'attente. Vous devez fournir:

- Nom du canal
- attribut Adresse
- Attribut SSLPEER, uniquement si le canal entrant utilise SSL ou TLS
- QMNAME, si le canal entrant est un canal de gestionnaire de files d'attente, ou
- CLNTUSER, si le canal entrant est un canal client

L'exemple suivant vérifie quelle règle CHLAUTH, avec les règles par défaut en place, permet à un `MQ-admin` utilisateur `johndoe` d'accéder à un canal nommé `CHAN1`:

```
runmqsc: DISPLAY CHLAUTH (CHAN1) MATCH (RUNCHECK) CLNTUSER ('johndoe') ADDRESS
('192.168.1.138')

AMQ8878: Display channel authentication record details.
CHLAUTH(*) TYPE(BLOCKUSER)
USERLIST(*MQADMIN)
```

Pour l'utilisateur `johndoe`, le canal ne s'exécute pas, l'utilisateur sera bloqué en raison de la règle BLOCKUSER pour les utilisateurs *MQADMIN.

L'exemple suivant vérifie quelle règle CHLAUTH, avec les règles par défaut en place, permet à l'utilisateur `alice` qui n'est pas un utilisateur `MQ-admin` d'accéder à un canal nommé `CHAN1`:

```
runmqsc: DISPLAY CHLAUTH (CHAN1) MATCH (RUNCHECK) CLNTUSER ('alice') ADDRESS
```

```
('192.168.1.138')

AMQ9783: Channel will run using MCAUSER('alice').
```

Pour l'utilisateur `alice`, le canal s'exécute et le canal transmet `alice` en tant que MCAUSER. MCAUSER est l'ID utilisateur utilisé pour vérifier les droits sur les objets IBM MQ .

Référence associée
SET CHLAUTH
AFFICHER CHLAUTH

Création de nouvelles règles CHLAUTH pour les utilisateurs
Quelques scénarios courants pour les utilisateurs et des exemples de règles CHLAUTH pour les exécuter.

Cette rubrique contient les scénarios suivants:

- «Contrôle de l'accès pour des utilisateurs MQ-admin spécifiques», à la page 64
- «Contrôle de l'accès pour un utilisateur spécifique et une application client IBM MQ», à la page 65
- «Contrôle de l'accès d'un utilisateur spécifique à l'aide du nom distinctif (DN) de certificat de cet utilisateur», à la page 65
- «Mappage d'un utilisateur particulier à l'utilisateur mqm», à la page 66

Contrôle de l'accès pour des utilisateurs MQ-admin spécifiques

Pour ce scénario, configurez un canal de connexion serveur qui doit être utilisé exclusivement pour une perspective d'administration, c'est-à-dire pour la connexion à partir de IBM MQ Explorer. Vous disposez d'un canal spécifique pour cette utilisation, et d'une ou de plusieurs adresses IP définies, à partir desquelles vous souhaitez que les connexions soient acceptées, et d'un accès bloqué pour l'ID `'mqm'` , si la connexion ne provient pas de l'une des adresses IP spécifiées.

Créez un canal SVRCONN pour les utilisateurs IBM MQ Explorer et `MQ-admin` appelés `ADMIN.CHAN`:

```
runmqsc: DEFINE CHANNEL (ADMIN.CHAN) CHLTYPE (SVRCONN) TRPTYPE (TCP)
```

A des fins de test, vérifiez que vous disposez d'un utilisateur défini dans le groupe `MQ-admin` et d'un autre utilisateur défini dans le groupe. Pour ce scénario, `mqadm` se trouve dans le groupe `MQ-admin` et `alice` ne se trouve pas dans le groupe.

Les règles CHLAUTH par défaut sont en place. Ajoutez trois règles pour permettre à un utilisateur spécifique d'accéder à ADMIN.CHAN en tant que `MQ-admin` à partir de certaines adresses IP:

- Définir NOACCESS à partir de n'importe quelle adresse
- Définissez BLOCKUSER pour ce canal afin de ne bloquer que l'utilisateur `nobody`, qui remplace *MQADMIN BLOCKUSER
- Accès ALLOW à l'utilisateur `mqadm` sur un sous-réseau spécifique d'adresses et MAP aux droits utilisateur `mqadm`

```
runmqsc:
SET CHLAUTH (ADMIN.CHAN) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
SET CHLAUTH('ADMIN.CHAN') TYPE(BLOCKUSER) +
DESCR('Rule to override *MQADMIN blockuser on this channel') +
USERLIST('nobody') ACTION(replace)
SET CHLAUTH('ADMIN.CHAN') TYPE(USERMAP) +
CLNTUSER('mqadm') USERSRC(MAP) MCAUSER('mqadm') +
ADDRESS('192.168.1.*') +
DESCR('Allow mqadm as mqadm on local subnet') ACTION(ADD)
```

A ce stade, l'utilisateur `mqadm` peut accéder à ADMIN.CHAN , à partir de la plage d'adresses IP spécifiée.

Vous pouvez exécuter MATCH (RUNCHECK) à tout moment pour afficher les résultats de chacune de ces commandes:

```
runmqsc:
DISPLAY CHLAUTH (ADMIN.CHAN) MATCH (RUNCHECK) CLNTUSER ('mqadm') ADDRESS
('192.168.1.138')
AMQ8878: Display channel authentication record details.
CHLAUTH(ADMIN.CHAN) TYPE(USERMAP)
ADDRESS(192.168.1.*) CLNTUSER(mqadm)
MCAUSER(mqadm)

DISPLAY CHLAUTH (ADMIN.CHAN) MATCH (RUNCHECK) CLNTUSER ('alice') ADDRESS
('192.168.1.138')
AMQ8878: Display channel authentication record details.
CHLAUTH(ADMIN.CHAN) TYPE(ADDRESSMAP)
ADDRESS(*) USERSRC(NOACCESS)
```

A ce stade, seuls les utilisateurs disposant d'un enregistrement CHLAUTH sont autorisés à accéder à l'aide de ADMIN.CHAN.

Contrôle de l'accès pour un utilisateur spécifique et une application client IBM MQ

Pour ce scénario, les règles CHLAUTH par défaut sont adéquates, en supposant que les droits IBM MQ doivent être définis pour un utilisateur spécifique, afin de fournir les droits IBM MQ appropriés (à l'aide de setmqaut).

Dans ce scénario, les droits sont définis pour un utilisateur `mqapp1`, qui n'est pas un utilisateur `MQ-admin` . Créez un canal SVRCONN, APP1.CHAN, à utiliser par une application particulière et un utilisateur spécifique.

```
runmqsc: DEFINE CHANNEL (APP1.CHAN) CHLTYPE (SVRCONN) TRPTYPE (TCP)
```

Lorsque les règles CHLAUTH par défaut sont en place, l'utilisateur `mqapp1` peut démarrer l'application APP1.CHAN .

L'ID utilisateur provenant de l'application client IBM MQ est utilisé pour la vérification des droits d'accès aux objets IBM MQ . Dans ce cas, en supposant que l'utilisateur'mqapp1'exécute l'application client IBM MQ , celle-ci est utilisée pour la vérification des droits d'accès aux objets IBM MQ . Par conséquent, si `mqapp1` a accès aux objets IBM MQ dont l'application a besoin, tout va bien ; si ce n'est pas le cas, vous obtiendrez des erreurs de droits d'accès.

Vous pouvez renforcer encore la sécurité en créant des règles CHLAUTH spécifiques pour l'ID utilisateur `mqapp1` , mais sous les règles par défaut, aucun membre du groupe `MQ-admin` ne peut accéder à ce canal.

```
runmqsc:
SET CHLAUTH (APP1.CHAN) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
SET CHLAUTH('APP1.CHAN') TYPE(USERMAP) +
CLNTUSER('mqapp1') USERSRC(MAP) MCAUSER('mqapp1') +
DESCR('Allow mqapp1 as mqapp1 on local subnet') ACTION(ADD)
```

Contrôle de l'accès d'un utilisateur spécifique à l'aide du nom distinctif (DN) de certificat de cet utilisateur

Pour ce scénario, l'utilisateur doit disposer d'un certificat transmis au gestionnaire de files d'attente. Le nom distinctif est ensuite comparé au paramètre SSLPEER de la règle CHLAUTH et SSLPEER peut utiliser des caractères génériques.

En cas de correspondance, l'utilisateur peut également être mappé à un autre utilisateur MCAUSER pour vérifier les droits sur les objets IBM MQ . Le mappage de MCAUSER peut réduire le nombre d'utilisateurs à gérer dans le gestionnaire des droits d'accès aux objets (OAM) IBM MQ .

Vous disposez d'un canal TLS avec des certificats en cours d'utilisation et vous avez besoin de règles pour:

- Bloquer tous les utilisateurs d'un canal particulier
- N'autorisez que les utilisateurs ayant un SSLPEER particulier qui utilisent le client de cet utilisateur pour l'accès à IBM MQ OAM.

```
.
# block all users on any IP address.
SET CHLAUTH('SSL1.SVRCONN') TYPE(ADDRESSMAP) ADDRESS('*')
USERSRC(NOACCESS) DESCR(''block all'') WARN(NO) ACTION(ADD)
.
# override - no MQM admin rule (allow mqm group /mqm admin users to
connect.
SET CHLAUTH('SSL1.SVRCONN') TYPE(BLOCKUSER) USERLIST('nobody')
DESCR('override no mqm admin rule') WARN(NO) ACTION(ADD)
.
# allow particular SSLPEER, use client id coming in from channel
SET CHLAUTH('SSL1.SVRCONN') TYPE(SSLPEERMAP)
SSLPEER('CN=JOHNDOE,O=IBM,C=US') USERSRC(CHANNEL) ACTION(ADD)
```

L'ID utilisateur client se connectant sur le canal est utilisé pour les droits IBM MQ OAM des objets IBM MQ ; par conséquent, l'ID utilisateur doit disposer des droits IBM MQ appropriés.

Vous pouvez effectuer un mappage vers un ID utilisateur IBM MQ différent si vous le souhaitez, à l'aide des éléments suivants:

```
USERSRC(MAP) MCAUSER('mquser1')
```

au lieu de USERSRC(CHANNEL).

Mappage d'un utilisateur particulier à l'utilisateur `mqm`

Il s'agit d'un ajout ou d'une modification à «Contrôle de l'accès pour des utilisateurs MQ-admin spécifiques», à la page 64.

Ajoutez la règle CHLAUTH suivante pour mapper des utilisateurs particuliers à l'utilisateur `mqm` , ou à un ID utilisateur `MQ-admin` , qui dispose des droits d'accès aux objets IBM MQ dans la méthode d'accès aux objets IBM MQ .

```
runmqsc:
SET CHLAUTH('ADMIN.CHAN') TYPE(USERMAP) +
CLNTUSER ('johndoe') USERSRC(MAP) MCAUSER ('mqm') +
ADDRESS('192.168.1-100.*') +
DESCR ('Allow johndoe as MQ-admin on local subnet') ACTION (ADD)
```

Cela permet et mappe l'utilisateur `johndoe` à l'utilisateur `mqm` pour le canal spécifique ADMIN.CHAN.

Concepts associés

«Résolution des problèmes d'accès CHLAUTH», à la page 62
Suggestions sur la façon de résoudre certains problèmes d'accès lors de l'utilisation d'enregistrements d'authentification de canal (CHLAUTH).

«Création de nouvelles règles CHLAUTH pour les canaux», à la page 66
Pour vous aider à créer vos propres règles CHLAUTH, voici quelques scénarios courants pour les canaux, ainsi que des exemples de règles CHLAUTH pour les exécuter.

Référence associée

SET CHLAUTH

AFFICHER CHLAUTH

Création de nouvelles règles CHLAUTH pour les canaux

Pour vous aider à créer vos propres règles CHLAUTH, voici quelques scénarios courants pour les canaux, ainsi que des exemples de règles CHLAUTH pour les exécuter.

Cette rubrique contient les scénarios suivants:

- «Autorisez uniquement l'accès à un canal particulier à partir d'une plage d'adresses IP spécifique.», à la page 67
- «Pour un canal spécifique, bloquez tous les utilisateurs, mais autorisez des utilisateurs spécifiques à se connecter.», à la page 67
- «Utilisation de CHLAUTH pour les canaux récepteur et émetteur», à la page 67

Autorisez uniquement l'accès à un canal particulier à partir d'une plage d'adresses IP spécifique.

Pour ce scénario, vous souhaitez:

- Aucun accès au canal depuis n'importe où
- Autoriser l'accès à partir d'une adresse IP ou d'une plage d'adresses IP spécifique

```
runmqsc:
SET CHLAUTH('APP2.CHAN') TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
WARN(NO) ACTION(ADD)
SET CHLAUTH('APP2.CHAN') TYPE(ADDRESSMAP) ADDRESS('9.95.100.1-5')
USERSRC(MAP) MCAUSER('mqapp2') ACTION(ADD)
```

Cela n'autorise que l'application APP2.CHAN à démarrer lorsque la connexion provient de la plage d'adresses IP spécifique spécifiée.

L'utilisateur se connectant en tant que MCAUSER est mappé à `mqapp2`et obtient par conséquent les droits OAM IBM MQ pour cet utilisateur.

Pour un canal spécifique, bloquez tous les utilisateurs, mais autorisez des utilisateurs spécifiques à se connecter.

Pour ce scénario, l'accès au canal MY.SVRCONN a les règles CHLAUTH par défaut en place.

Vous devez ajouter les éléments suivants:

```
# block all users
SET CHLAUTH('MY.SVRCONN') TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
DESCR(''block all'') WARN(NO) ACTION(ADD)

# override - no MQM admin rule
SET CHLAUTH('MY.SVRCONN') TYPE(BLOCKUSER) USERLIST('nobody') DESCR('override
no mqm admin rule') WARN(NO) ACTION(ADD)

# allow johndoe userid
SET CHLAUTH('MY.SVRCONN') TYPE(USERMAP) CLNTUSER('johndoe')
USERSRC(CHANNEL) DESCR('allow johndoe userid') ACTION(ADD)
```

Cette première partie du code empêche toute personne de se connecter à MY.SVRCONN, puis le code autorise uniquement le canal MY.SVRCONN à être démarré lorsque la connexion provient de l'ID utilisateur spécifique `johndoe`.

L'utilisateur se connectant sur le canal `johndoe` est utilisé pour les droits IBM MQ OAM des objets IBM MQ . Par conséquent, l'ID utilisateur doit disposer des droits IBM MQ appropriés.

Vous pouvez effectuer un mappage vers un ID utilisateur IBM MQ différent si vous le souhaitez, à l'aide des éléments suivants:

```
USERSRC(MAP) MCAUSER('mquser1')
```

au lieu de `USERSRC(CHANNEL)`.

Utilisation de CHLAUTH pour les canaux récepteur et émetteur

Vous pouvez utiliser les règles CHLAUTH pour ajouter une sécurité supplémentaire aux canaux récepteur et émetteur, afin de restreindre l'accès au canal récepteur. Notez que si vous ajoutez ou modifiez des

règles CHLAUTH, les règles CHLAUTH mises à jour ne s'appliquent qu'au démarrage du canal. Par conséquent, si les canaux sont déjà en cours d'exécution, vous devez les arrêter et les redémarrer pour que les mises à jour CHLAUTH s'appliquent.

Les règles HLAUTH peuvent être utilisées sur n'importe quel canal, mais il existe certaines restrictions. Par exemple, les règles USERMAP s'appliquent uniquement aux canaux SVRCONN.

Cet exemple permet une connexion à partir d'une adresse IP particulière uniquement, pour démarrer TO.MYSVR1 :

```
# First you could lock down the channel by disallowing all
# for channel 'TO.MYSVR1', RCVR channel
SET CHLAUTH('TO.MYSVR1') TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
DESCR('Back-stop rule')

# Then you could allow this channel to be started
SET CHLAUTH('TO.MYSVR1') TYPE(ADDRESSMAP) ADDRESS('192.168.1.134') USERSRC(MAP)
MCAUSER('mqapp') ACTION(ADD)
```

Cet exemple permet la connexion à partir d'un gestionnaire de files d'attente particulier uniquement:

```
# Lock down all access:
SET CHLAUTH('TO.MYSVR1') TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS)
DESCR('Back-stop rule')

# Then allow access from queue manager MYSVR2 and from a particular ipaddress:
SET CHLAUTH('TO.MYSVR1') TYPE(QMGRMAP) QMNAME('MYSVR2') USERSRC(MAP)
MCAUSER('mqapp') ADDRESS('192.168.1.134') ACTION(ADD)
```

Concepts associés

«Résolution des problèmes d'accès CHLAUTH», à la page 62
Suggestions sur la façon de résoudre certains problèmes d'accès lors de l'utilisation d'enregistrements d'authentification de canal (CHLAUTH).

«Création de nouvelles règles CHLAUTH pour les utilisateurs», à la page 64
Quelques scénarios courants pour les utilisateurs et des exemples de règles CHLAUTH pour les exécuter.

Référence associée

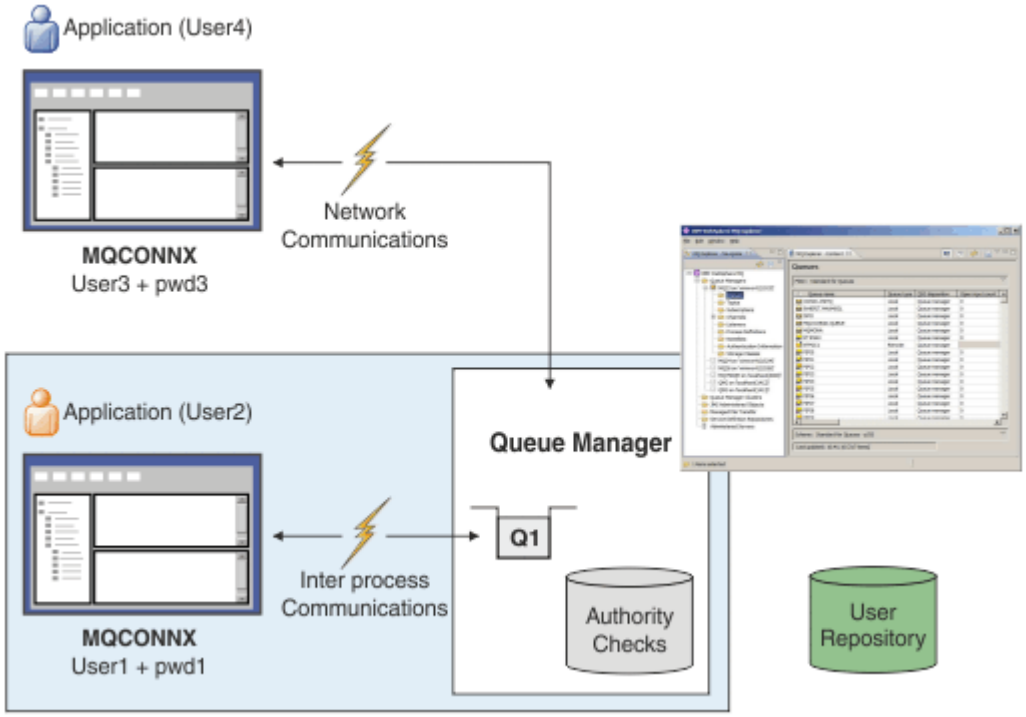
SET CHLAUTH

AFFICHER CHLAUTH

Authentification de connexion

L'authentification de connexion peut être effectuée de différentes manières:

- Une application peut fournir un ID utilisateur et un mot de passe. L'application peut être un client ou utiliser des liaisons locales.
- Un gestionnaire de files d'attente peut être configuré pour agir sur un ID utilisateur et un mot de passe fournis.
- Un référentiel peut être utilisé pour déterminer si une combinaison d'ID utilisateur et de mot de passe est valide.



Dans le diagramme, deux applications établissent des connexions avec un gestionnaire de files d'attente, une application en tant que client et une autre utilisant des liaisons locales. Les applications peuvent utiliser une variété d'API pour se connecter au gestionnaire de files d'attente, mais toutes ont la possibilité de fournir un ID utilisateur et un mot de passe. L'ID utilisateur sous lequel l'application s'exécute, `User2` et `User4` , dans le diagramme, qui est l'ID utilisateur de système d'exploitation habituel présenté à IBM MQ, peut être différent de l'ID utilisateur fourni par l'application, `User1` et `User3`.

Le gestionnaire de files d'attente reçoit les commandes de configuration (dans le diagramme, IBM MQ Explorer est utilisé) et gère l'ouverture des ressources et vérifie les droits d'accès à ces ressources. Il existe de nombreuses ressources différentes dans IBM MQ auxquelles une application peut avoir besoin de droits d'accès. Le diagramme illustre l'ouverture d'une file d'attente pour la sortie, mais les mêmes principes s'appliquent également aux autres ressources.

Voir Référentiels d'utilisateurs pour plus de détails sur le référentiel utilisé pour vérifier les ID utilisateur et les mots de passe.

Concepts associés

«Authentification de connexion: Configuration», à la page 69
Un gestionnaire de files d'attente peut être configuré pour utiliser un ID utilisateur et un mot de passe fournis afin de vérifier si un utilisateur dispose des droits d'accès aux ressources.

«Authentification de la connexion: modifications de l'application», à la page 73

«Authentification de connexion: référentiels d'utilisateurs», à la page 74
Pour chacun de vos gestionnaires de files d'attente, vous pouvez choisir différents types d'objet d'informations d'authentification pour l'authentification des ID utilisateur et des mots de passe.

Authentification de connexion: Configuration

Un gestionnaire de files d'attente peut être configuré pour utiliser un ID utilisateur et un mot de passe fournis afin de vérifier si un utilisateur dispose des droits d'accès aux ressources.

Activation de l'authentification de connexion sur un gestionnaire de files d'attente

Sur un objet gestionnaire de files d'attente, l'attribut **CONNAUTH** peut être défini sur le nom d'un objet d'informations d'authentification (AUTHINFO). Cet objet peut être de deux types (attribut AUTHTYPE):

IDPWOS
Indique que le gestionnaire de files d'attente utilise le système d'exploitation local pour authentifier l'ID utilisateur et le mot de passe.

IDPWLDAP
Indique que le gestionnaire de files d'attente utilise un serveur LDAP pour authentifier l'ID utilisateur et le mot de passe.

Remarque : Vous ne pouvez pas utiliser d'autre type d'objet d'informations d'authentification dans la zone **CONNAUTH** .

IDPWOS et IDPWLDAP sont similaires dans un certain nombre de leurs attributs, qui sont décrits ici. D'autres attributs sont pris en compte ultérieurement.

Pour vérifier les connexions locales, utilisez l'attribut AUTHINFO **CHCKLOCL** (vérifier les connexions locales). Pour vérifier les connexions client, utilisez l'attribut AUTHINFO **CHCKCLNT** (vérifier les connexions client). La configuration doit être actualisée avant que le gestionnaire de files d'attente ne reconnaisse les modifications.

```
ALTER QMGR CONNAUTH(USE.PW)
DEFINE AUTHINFO(USE.PW) +
AUTHTYPE(IDPWOS) +
FAILDLAY(10) +
CHCKLOCL(OPTIONAL) +
CHCKCLNT(REQUIRED)
REFRESH SECURITY TYPE(CONNAUTH)
```

Où USE.PW dans CONNAUTH est une chaîne qui correspond à la définition AUTHINFO.

CHCKLOCL et **CHCKCLNT** ont le même ensemble de valeurs possibles qui permettent de varier la rigueur de la vérification:

Aucun
Désactive la vérification.

Facultatif
S'assure que si un ID utilisateur et un mot de passe sont fournis par une application, ils constituent une paire valide, mais qu'il n'est pas obligatoire de les fournir. Cette option peut être utile lors de la migration, par exemple.

Important : OPTIONAL est la valeur minimale que vous pouvez définir, afin d'utiliser des règles CHLAUTH plus strictes.

Si vous sélectionnez NONE et que la connexion client correspond à un enregistrement CHLAUTH avec CHCKCLNT REQUIRED (ou REQDADM sur les plateformes autres que z/OS), la connexion échoue. Vous recevez le message AMQ9793 sur les plateformes autres que z/OSet le message CSQX793E sur z/OS.

required
Exige que toutes les applications fournissent un ID utilisateur et un mot de passe valides. Voir aussi la remarque suivante.

REQDADM
Les utilisateurs privilégiés doivent fournir un ID utilisateur et un mot de passe valides, mais les utilisateurs non privilégiés sont traités comme avec le paramètre OPTIONAL . Voir aussi la remarque suivante. z/OS (Ce paramètre n'est pas autorisé sur les systèmes z/OS .)

Remarque :

La définition de **CHCKLOCL** sur REQUIRED ou REQDADM signifie que vous ne pouvez pas administrer localement le gestionnaire de files d'attente à l'aide de **runmqsc** (erreur AMQ8135: Non autorisé) sauf si l'utilisateur spécifie le paramètre -u UserId sur la ligne de commande **runmqsc** . Lorsque cette option est définie, **runmqsc** vous invite à indiquer le mot de passe de l'utilisateur sur la console.

De même, un utilisateur exécutant IBM MQ Explorer sur le système local verra l'erreur AMQ4036 lors de la tentative de connexion au gestionnaire de files d'attente. Pour spécifier un nom d'utilisateur et un mot de passe, cliquez avec le bouton droit de la souris sur l'objet du gestionnaire de files d'attente local et

sélectionnez **Détails de connexion** > **Propriétés ...** dans le menu. Dans la section **ID utilisateur** , entrez le nom d'utilisateur et le mot de passe à utiliser, puis cliquez sur **OK**.

Des considérations similaires s'appliquent aux connexions distantes avec **CHCKCLNT**.

CONNAUTH est vide pour les gestionnaires de files d'attente migrés mais défini sur *SYSTEM.DEFAULT.AUTHINFO.IDPWOS* pour les nouveaux gestionnaires de files d'attente. La définition **AUTHINFO** précédente a **CHCKCLNT** défini sur *REQDADM* par défaut.

Par conséquent, vous devez fournir le mot de passe de système d'exploitation correct pour tous les clients existants à l'aide d'un ID utilisateur privilégié pour la connexion.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont protégés de manière appropriée, voir «Protection par mot de passe MQCSP», à la page 30.

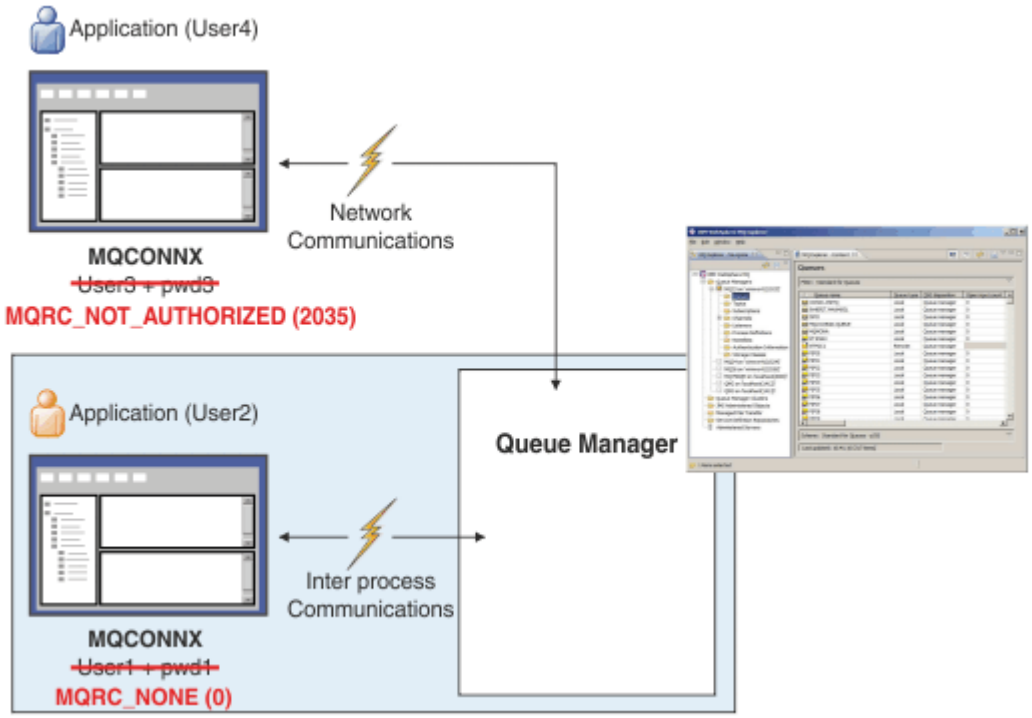
Granularité de la configuration

Outre **CHCKLOCL** et **CHCKCLNT** qui sont utilisés pour activer la vérification de l'ID utilisateur et du mot de passe, des améliorations ont été apportées aux règles CHLAUTH afin que des configurations plus spécifiques puissent être effectuées à l'aide de **CHCKCLNT**.

Vous pouvez définir la valeur **CHCKCLNT** globale sur OPTIONAL, par exemple, puis la mettre à niveau pour qu'elle soit plus stricte pour certains canaux en définissant **CHCKCLNT** sur REQUIRED ou REQDADM sur la règle CHLAUTH . Par défaut, les règles CHLAUTH s'exécutent avec CHCKCLNT(ASQMGR) de sorte que cette granularité n'a pas besoin d'être utilisée. Exemple :

```
DEFINE AUTHINFO(USE.PW) AUTHTYPE(xxxxxx) +
CHCKCLNT(OPTIONAL)
SET CHLAUTH('*') TYPE(ADDRESSMAP) +
ADDRESS('*') USERSRC(CHANNEL) +
CHCKCLNT(REQUIRED)
SET CHLAUTH('*') TYPE(SSLPEERMAP) +
SSLPEER('CN=*') USERSRC(CHANNEL)
```

Notification d'erreur



Une erreur est enregistrée si une application ne fournit pas d'ID utilisateur et de mot de passe lorsqu'elle est requise ou si elle fournit une combinaison incorrecte même si elle est facultative.

Remarque : Lorsque la vérification des mots de passe est désactivée, si vous utilisez l'option NONE sur **CHCKLOCL** ou **CHCKCLNT**, les mots de passe non valides ne sont pas détectés.

Les échecs d'authentification sont conservés pendant le nombre de secondes spécifié par l'attribut **FAILDLAY** avant que l'erreur ne soit renvoyée à l'application. Cela offre une protection contre les tentatives répétées de connexion d'une application.

L'erreur est enregistrée de plusieurs manières:

Filtre
L'application renvoie l'erreur de sécurité IBM MQ standard, RC2035 -MQRC_NOT_AUTHORIZED.

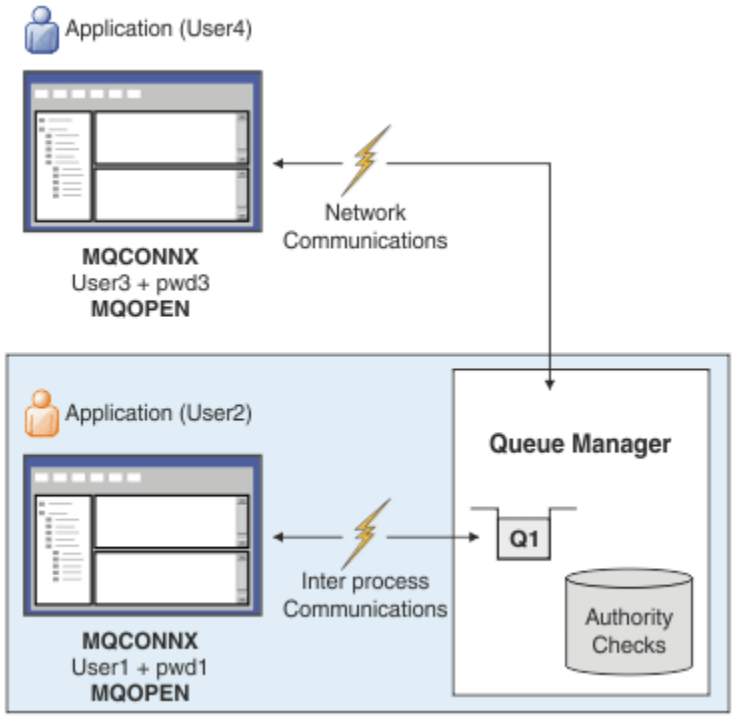
Administrateur
Un administrateur IBM MQ voit l'événement signalé dans le journal des erreurs et peut donc voir que l'application a été rejetée car la vérification de l'ID utilisateur et du mot de passe a échoué, plutôt que parce que, par exemple, il n'y avait pas de droits de connexion .

Outil de surveillance
Un outil de surveillance peut également être averti de l'échec, si vous activez les événements de droits d'accès en envoyant un message d'événement à SYSTEM.ADMIN.QMGR.EVENT EVENT:

```
ALTER QMGR AUTHOREV(ENABLED)
```

Cet événement "Non autorisé" est un événement de connexion de type 1 et fournit les mêmes zones que les autres événements de type 1, avec une zone supplémentaire, l'ID utilisateur MQCSP fourni. Le mot de passe n'est pas indiqué dans le message d'événement. Cela signifie que le message d'événement contient deux ID utilisateur: l'ID sous lequel l'application s'exécute et l'ID que l'application a présenté pour la vérification de l'ID utilisateur et du mot de passe.

Relation à l'autorisation



Vous pouvez configurer un gestionnaire de files d'attente pour exiger que les ID utilisateur et les mots de passe soient fournis par certaines applications car l'ID utilisateur sous lequel l'application s'exécute peut ne pas être le même que celui présenté par l'application avec un mot de passe lorsque l'application ouvre une file d'attente pour la sortie, par exemple:

```
ALTER QMGR CONNAUTH(USE.PWD)
DEFINE AUTHINFO(USE.PWD) +
AUTHTYPE(xxxxxx) +
CHCKLOCL(OPTIONAL) +
CHCKCLNT(REQUIRED) +
ADOPTCTX(YES)
```

La façon dont les ID utilisateur et les mots de passe sont gérés est contrôlée par l'attribut **ADOPTCTX** sur l'objet d'informations d'authentification.

ADOPTCTX (OUI)
Toutes les vérifications d'autorisation d'une application sont effectuées avec le même ID utilisateur que celui que vous avez authentifié par mot de passe, en choisissant d'adopter le contexte comme contexte d'application pour le reste de la durée de vie de la connexion.

Avertissement : Lorsque vous utilisez ADOPTCTX (YES) et les ID utilisateur du système d'exploitation, vous devez vous assurer que l'ID utilisateur adopté ne dépasse pas la longueur maximale des ID utilisateur. Pour plus d'informations, voir «ID utilisateur», à la page 84.

ADOPTCTX (NON)
Une application fournit un ID utilisateur et un mot de passe pour les authentifier lors de la connexion, mais elle continue en utilisant l'ID utilisateur sous lequel l'application s'exécute pour les vérifications d'autorisation ultérieures. Cette option peut s'avérer utile lors de la migration ou si vous prévoyez d'utiliser d'autres mécanismes, tels que des enregistrements d'authentification de canal, pour affecter l' ID utilisateur de l'agent de canal de message (MCAUSER).

Avertissement :

Si vous utilisez le paramètre **ADOPTCTX(YES)** sur un objet d'informations d'authentification, vous ne pouvez pas adopter un autre contexte de sécurité à moins que vous ne définissiez le paramètre **ChlauthEarlyAdopt** dans la strophe channels du fichier `qm.ini`.

Par exemple, l'objet d'informations d'authentification par défaut est défini sur **ADOPTCTX(YES)** et l'utilisateur `fred` est connecté. Les deux règles CHLAUTH suivantes sont configurées:

```
SET CHLAUTH('MY.CHLAUTH') TYPE(ADDRESSMAP) DESCR('Block all access by
default') ADDRESS('*') USERSRC(NOACCESS) ACTION(REPLACE)
SET CHLAUTH('MY.CHLAUTH') TYPE(USERMAP) DESCR('Allow user bob and force
CONNAUTH') CLNTUSER('bob') CHCKCLNT(REQUIRED) USERSRC(CHANNEL)
```

La commande suivante est émise, avec l'intention d'authentifier la commande en tant que contexte de sécurité adopté de l'utilisateur bob:

```
runmqsc -c -u bob QMGR
```

En fait, le gestionnaire de files d'attente utilise le contexte de sécurité `fred`, et non bob, et la connexion échoue.

Pour plus d'informations sur **ChlauthEarlyAdopt**, voir Attributs de la strophe channels.

Concepts associés

«Authentification de connexion», à la page 68

«Authentification de la connexion: modifications de l'application», à la page 73

«Authentification de connexion: référentiels d'utilisateurs», à la page 74

Pour chacun de vos gestionnaires de files d'attente, vous pouvez choisir différents types d'objet d'informations d'authentification pour l'authentification des ID utilisateur et des mots de passe.

Authentification de la connexion: modifications de l'application

Une application peut fournir un ID utilisateur et un mot de passe dans la structure des paramètres de sécurité de connexion (MQCSP) lorsque MQCONNX est appelé. L'ID utilisateur et le mot de passe sont transmis pour vérification au gestionnaire des droits d'accès aux objets (OAM) fourni avec le gestionnaire

de files d'attente ou au composant de service d'autorisation fourni avec le gestionnaire de files d'attente sur les systèmes z/OS . Vous n'avez pas besoin d'écrire votre propre interface personnalisée.

Si l'application s'exécute en tant que client, l'ID utilisateur et le mot de passe sont également transmis aux exits de sécurité côté client et côté serveur pour traitement. Ils peuvent également être utilisés pour définir l' attribut d'ID utilisateur d'agent de canal de message (MCAUSER) d'une instance de canal. L'exit de sécurité est appelé avec le motif d'exit MQXR_SEC_PARMS pour ce traitement. Les exits de sécurité côté client et l'exit de préconnexion peuvent apporter des modifications à MQCONN avant son envoi au gestionnaire de files d'attente.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont protégés de manière appropriée, voir «Protection par mot de passe MQCSP», à la page 30.

En utilisant la chaîne XAOPEN pour fournir un ID utilisateur et un mot de passe, vous pouvez éviter d'avoir à modifier le code de l'application.

Remarque :

Depuis la IBM WebSphere MQ 6.0, l'exit de sécurité a autorisé la définition du MQCSP. Par conséquent, les clients de ce niveau ou d'un niveau ultérieur n'ont pas besoin d'être mis à niveau.

Toutefois, dans les versions de IBM MQ antérieures à IBM MQ 8.0, MQCSP ne plaçait aucune restriction sur l'ID utilisateur et le mot de passe fournis par l'application. Lorsque vous utilisez ces valeurs avec des fonctions fournies par IBM MQ , il existe des limites qui s'appliquent à l'utilisation de ces fonctions, mais si vous les transmettez uniquement à vos propres exits, ces limites ne s'appliquent pas.

Concepts associés

«Authentification de connexion», à la page 68

«Authentification de connexion: Configuration», à la page 69
Un gestionnaire de files d'attente peut être configuré pour utiliser un ID utilisateur et un mot de passe fournis afin de vérifier si un utilisateur dispose des droits d'accès aux ressources.

«Authentification de connexion: référentiels d'utilisateurs», à la page 74
Pour chacun de vos gestionnaires de files d'attente, vous pouvez choisir différents types d'objet d'informations d'authentification pour l'authentification des ID utilisateur et des mots de passe.

Authentification de connexion: référentiels d'utilisateurs

Pour chacun de vos gestionnaires de files d'attente, vous pouvez choisir différents types d'objet d'informations d'authentification pour l'authentification des ID utilisateur et des mots de passe.

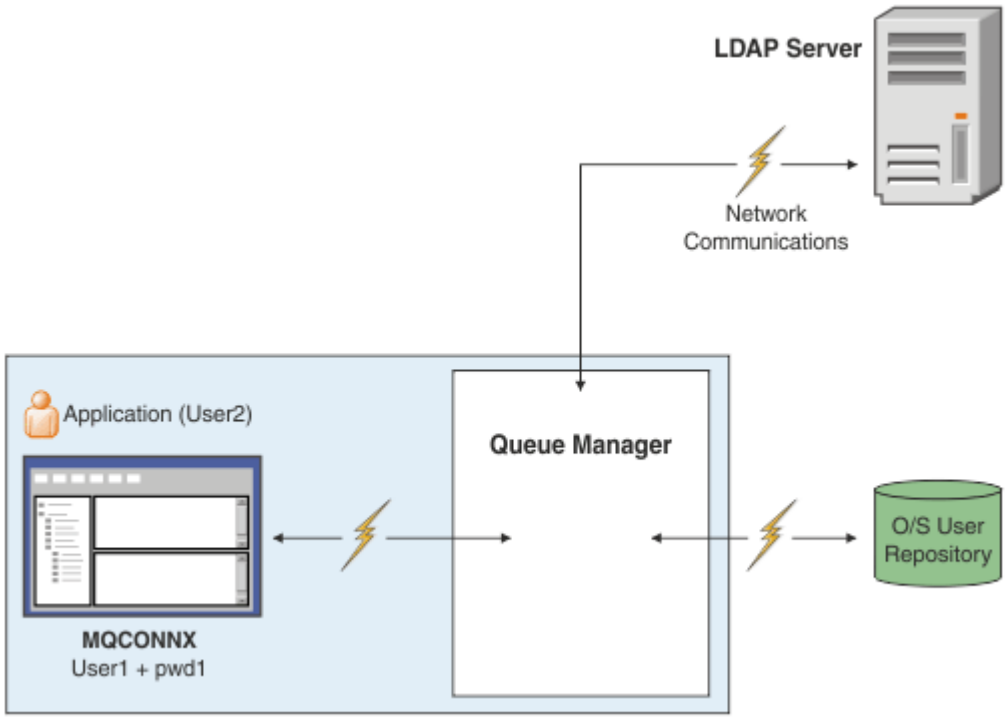


Figure 7. Types d'objets d'informations d'authentification

```
DEFINE AUTHINFO(USE.OS) AUTHTYPE(IDPWOS)
DEFINE AUTHINFO(USE.LDAP) +
AUTHTYPE(IDPWLDAP) +
CONNAME('ldap1(389),ldap2(389)') +
LDAPUSER('CN=QMGR1') +
LDAPPWD('passw0rd') SECCOMM(YES)
```

Il existe deux types d'objet d'informations d'authentification, comme représenté dans le diagramme:

- `IDPWOS` est utilisé pour indiquer que le gestionnaire de files d'attente utilise le système d'exploitation local pour authentifier l'ID utilisateur et le mot de passe. Si vous choisissez d'utiliser le système d'exploitation local, vous devez définir les attributs communs, comme décrit dans les rubriques précédentes.
- `IDPWLDAP` est utilisé pour indiquer que le gestionnaire de files d'attente utilise un serveur LDAP pour authentifier l'ID utilisateur et le mot de passe. Si vous choisissez d'utiliser un serveur LDAP, des informations supplémentaires sont fournies dans cette rubrique.

Un seul type d'objet d'informations d'authentification peut être choisi pour chaque gestionnaire de files d'attente à utiliser, en nommant l'objet approprié dans l'attribut **CONNAUTH** du gestionnaire de files d'attente.

Utilisation d'un serveur LDAP pour l'authentification.

Définissez la zone **CONNAME** sur l'adresse du serveur LDAP pour le gestionnaire de files d'attente. Vous pouvez fournir des adresses supplémentaires pour le serveur LDAP dans une liste séparée par des virgules, ce qui peut faciliter la redondance si le serveur LDAP ne fournit pas cette fonction lui-même.

Définissez l'ID et le mot de passe du serveur LDAP requis dans les zones **LDAPUSER** et **LDAPPWD** afin que le gestionnaire de files d'attente puisse accéder au serveur LDAP et rechercher des informations sur les enregistrements utilisateur.

Connexion sécurisée à un serveur LDAP

Contrairement aux canaux, il n'existe pas de paramètre **SSLCIPH** pour activer l'utilisation de TLS pour la communication avec le serveur LDAP. Dans ce cas, IBM MQ agit en tant que client sur le serveur LDAP et une grande partie de la configuration est effectuée sur le serveur LDAP. Certains paramètres existants dans IBM MQ sont utilisés pour configurer le fonctionnement de cette connexion.

Définissez la zone **SECCOMM** pour contrôler si la connectivité au serveur LDAP utilise TLS.

En plus de cet attribut, les attributs de gestionnaire de files d'attente **SSLFIPS** et **SUITEB** restreignent l'ensemble des spécifications de chiffrement qui sont choisies. Le certificat utilisé pour identifier le gestionnaire de files d'attente sur le serveur LDAP est le certificat du gestionnaire de files d'attente, `ibmwebspheremq` *`qmgr-name`* ou la valeur de l'attribut **CERTLABL** . Pour plus de détails voir Labels de certificat numérique.

Référentiel d'utilisateurs LDAP

Lors de l'utilisation d'un référentiel d'utilisateurs LDAP, d'autres opérations de configuration doivent être effectuées sur le gestionnaire de files d'attente, mais pas uniquement pour indiquer au gestionnaire de files d'attente où trouver le serveur LDAP.

Les ID utilisateur définis dans un serveur LDAP possèdent une structure hiérarchique qui les identifie de manière unique. Par conséquent, une application peut se connecter au gestionnaire de files d'attente et présenter son ID utilisateur en tant qu'ID utilisateur hiérarchique qualifié complet.

Toutefois, pour simplifier les informations qu'une application doit fournir, il est possible de configurer le gestionnaire de files d'attente pour qu'il considère que la première partie de la hiérarchie est commune à tous les ID et de l'ajouter automatiquement avant l'ID abrégé fourni par l'application. Le gestionnaire de files d'attente peut alors présenter un ID complet au serveur LDAP.

Définissez BASEDNU sur le point initial où la recherche LDAP recherche l'ID dans la hiérarchie LDAP. Lorsque vous définissez BASEDNU, vous devez vous assurer qu'un seul résultat est renvoyé lorsque vous recherchez l'ID dans la hiérarchie LDAP.

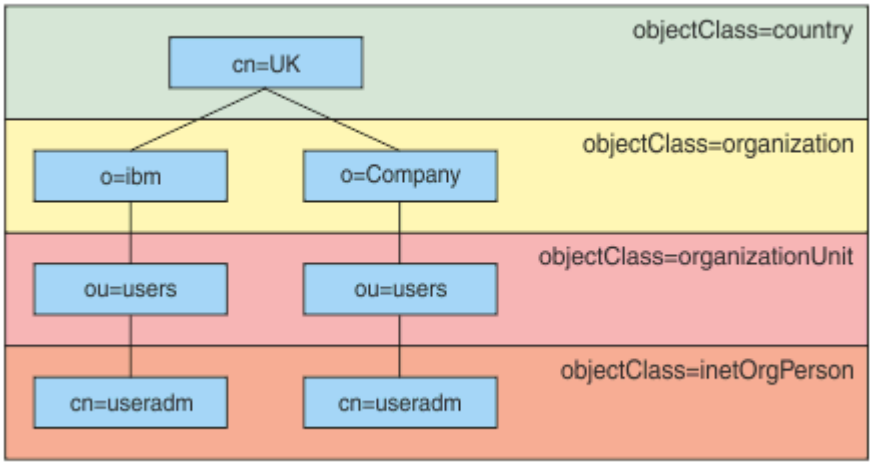


Figure 8. Exemple de hiérarchie LDAP

Par exemple, dans Figure 8, à la page 76 BASEDNU, la valeur peut être "ou=users, o=ibm, c = UK" ou ", o=ibm, c = UK". Cependant, comme un nom distinctif contenant "cn = useradm" existe à la fois dans la branche "o = ibm" et dans la branche "o=Company", BASEDNU ne peut pas être défini sur "c = UK". Pour des raisons de performances et de sécurité, utilisez le point le plus élevé de votre hiérarchie LDAP à partir duquel vous pouvez référencer tous les ID utilisateur dont vous avez besoin. Dans cet exemple, il s'agit de "ou=users, o=ibm, c = UK".

Votre application peut soumettre au gestionnaire de files d'attente l'ID utilisateur sans fournir le nom d'attribut LDAP, CN= par exemple. Si vous définissez USRFIELD sur le nom d'attribut LDAP, cette valeur

est ajoutée en tant que préfixe à l'ID utilisateur provenant de l'application. Il peut s'agir d'une aide à la migration utile lorsque vous passez des ID utilisateur du système d'exploitation aux ID utilisateur LDAP, car l'application peut alors présenter la même chaîne dans les deux cas et vous pouvez éviter de modifier l'application.

Par conséquent, l'ID utilisateur complet présenté au serveur LDAP se présente comme suit:

```
USRFIELD = ID_from_application BASEDNU
```

Concepts associés

«Authentification de connexion», à la page 68

«Authentification de connexion: Configuration», à la page 69
Un gestionnaire de files d'attente peut être configuré pour utiliser un ID utilisateur et un mot de passe fournis afin de vérifier si un utilisateur dispose des droits d'accès aux ressources.

«Authentification de la connexion: modifications de l'application», à la page 73

Exit de sécurité côté client pour l'insertion d'un ID utilisateur et d'un mot de passe (`mqccred`)

Si vous disposez d'applications client qui sont requises pour envoyer un ID utilisateur ou un mot de passe mais que vous ne pouvez pas encore modifier la source, un exit de sécurité fourni avec IBM MQ 8.0 appelé **`mqccred`** est disponible. **`mqccred`** fournit un ID utilisateur et un mot de passe pour le compte de l'application client, à partir d'un fichier `.ini` . Cet ID utilisateur et ce mot de passe sont envoyés au gestionnaire de files d'attente qui, s'il est configuré pour le faire, les authentifiera.

Présentation

`mqccred` est un exit de sécurité qui s'exécute sur la même machine que votre application client. Il permet de fournir des informations d'ID utilisateur et de mot de passe pour le compte de l'application client, lorsque ces informations ne sont pas fournies par l'application elle-même. Les informations d'ID utilisateur et de mot de passe sont fournies dans une structure appelée Connection Security Parameters (MQCSP) et sont authentifiées par le gestionnaire de files d'attente si l' authentification de connexion est configurée.

Les informations d'ID utilisateur et de mot de passe sont extraites d'un fichier `.ini` sur la machine client. Les mots de passe du fichier sont protégés par le brouillage à l'aide de la commande **`runmqccred`** et en s'assurant que les droits d'accès au fichier `.ini` sont définis de sorte que seul l'ID utilisateur exécutant l'application client (et donc l'exit) puisse le lire.

Emplacement

`mqccred` est installé:

Windows plateformes
Dans le répertoire *`installation_directory`*`\Tools\c\Samples\mqccred\`

UNIX plateformes
Dans le répertoire *`installation_directory`*`/samp/mqccred`

Remarques : L'exit:

1. Agit uniquement comme un exit de canal de sécurité et doit être le seul exit de ce type défini sur un canal.
2. Est généralement nommé via la table de définition de canal du client (CCDT), mais un client Java peut avoir directement l'exit mentionné dans les objets JNDI, ou l'exit peut être configuré pour les applications qui construisent manuellement la structure MQCD .
3. Vous devez copier les programmes **`mqccred`** et **`mqccred_r`** dans le répertoire `var/mqm/exits` .

Par exemple, sur une machine de plateforme UNIX 64 bits, exécutez la commande suivante:

```
cp installation_directory/samp/mqccred/lib64/* /var/mqm/exits
```

Pour plus d'informations, voir Exemple étape par étape de test de mqccred .

4. Est capable de s'exécuter sur des versions précédentes d' IBM MQ, aussi anciennes que IBM WebSphere MQ 7.0.1.

Configuration des ID utilisateur et des mots de passe

Le fichier `.ini` contient des sections pour chaque gestionnaire de files d'attente, avec un paramètre global pour les gestionnaires de files d'attente non spécifiés. Chaque section contient le nom du gestionnaire de files d'attente, un ID utilisateur et un texte en clair ou un mot de passe brouillé.

Vous devez éditer le fichier `.ini` à la main, à l'aide de l'éditeur de votre choix, et ajouter l'attribut de mot de passe en texte en clair aux sections. Exécutez le programme **runmqccred** fourni, qui utilise le fichier `.ini` et remplace l'attribut **Password** par l'attribut **OPW** , une forme brouillée du mot de passe.

Voir runmqccred pour une description de la commande et de ses paramètres.

Le fichier `mqccred.ini` contient vos informations d'ID utilisateur et de mot de passe.

Un modèle de fichier `.ini` est fourni dans le même répertoire que l'exit pour fournir un point de départ à votre entreprise.

Par défaut, ce fichier est recherché dans $HOME/`.mqs/mqccred.ini`. Si vous souhaitez le localiser ailleurs, vous pouvez utiliser la variable d'environnement *MQCCRED* pour le localiser:

```
MQCCRED=C:\mydir\mqccred.ini
```

Si vous utilisez *MQCCRED*, la variable doit inclure le nom complet du fichier de configuration, y compris tout type de fichier `.ini` . Etant donné que ce fichier contient des mots de passe (même s'ils sont brouillés), vous devez protéger le fichier à l'aide des privilèges du système d'exploitation pour vous assurer que les personnes non autorisées ne peuvent pas le lire. Si vous ne disposez pas des droits d'accès au fichier appropriés, l'exit ne s'exécutera pas correctement.

Si l'application a déjà fourni une structure MQCSP , l'exit le respecte normalement et n'insère aucune information du fichier `.ini` . Toutefois, vous pouvez le remplacer à l'aide de l'attribut **Force** de la section.

La définition de **Force** sur la valeur *TRUE* supprime l'ID utilisateur et le mot de passe fournis par l'application et les remplace par la version du fichier ini.

Vous pouvez également définir l'attribut **Force** dans la section globale du fichier pour définir la valeur par défaut de ce fichier.

La valeur par défaut de **Force** est *FALSE*.

Vous pouvez fournir un ID utilisateur et un mot de passe pour tous les gestionnaires de files d'attente ou pour chaque gestionnaire de files d'attente individuel. Voici un exemple de fichier `mqccred.ini` :

```
# comments are permitted
AllQueueManagers:
User=abc
OPW=%^&aervrgtsr

QueueManager:
Name=QMA
User=user1
OPW=H&^dbgfh

Force=TRUE

QueueManager:
Name=QMB
User=user2
password=passw0rd
```

Remarques :

1. Les définitions de gestionnaire de files d'attente individuelles sont prioritaires sur le paramètre global.
2. Les attributs sont insensibles à la casse.

Contraintes

Lorsque cet exit est utilisé, l'ID utilisateur local de la personne exécutant l'application n'est pas transmis du client au serveur. Les seules informations d'identité disponibles proviennent du contenu du fichier ini.

Par conséquent, vous devez configurer le gestionnaire de files d'attente pour qu'il utilise **ADOPTCTX(YES)** ou mapper la demande de connexion entrante à un ID utilisateur approprié via l'un des mécanismes disponibles, par exemple, «Enregistrements d'authentification de canal», à la page 50.

Important : Si vous ajoutez de nouveaux mots de passe ou mettez à jour les anciens, la commande **runmqccred** ne traite que les mots de passe en texte en clair, sans toucher à ceux qui sont brouillés.

Débogage

L'exit écrit dans la trace IBM MQ standard lorsqu'elle est activée.

Pour faciliter le débogage des problèmes de configuration, l'exit peut également écrire directement dans stdout.

Aucune donnée d'exit de sécurité de canal (**SCYDATA**) la configuration est normalement requise pour le canal. Toutefois, vous pouvez spécifier:

erreur
Seules les informations d'impression sont associées à des conditions d'erreur, telles que l'impossibilité de trouver le fichier de configuration.

DEBOGAGE
Affiche ces conditions d'erreur et des instructions de trace supplémentaires.

NOCHECKS
Ignore les contraintes sur les droits d'accès aux fichiers et la contrainte supplémentaire selon laquelle le fichier `.ini` ne doit pas contenir de mots de passe non protégés.

Vous pouvez placer un ou plusieurs de ces éléments dans la zone **SCYDATA** , séparés par des virgules, dans n'importe quel ordre. Par exemple, `SCYDATA=(NOCHECKS,DEBUG)`.

Notez que les éléments sont sensibles à la casse et doivent être entrés en majuscules.

Utilisation `mqccred`

Une fois votre fichier configuré, vous pouvez appeler l'exit de canal en mettant à jour votre définition de canal de connexion client pour inclure l'attribut `SCYEXIT('mqccred(ChlExit)')` :

```
DEFINE CHANNEL(channelname) CHLTYPE(clntconn) +
CONNAME(remote machine) +
QMNAME(remote qmgr) +
SCYEXIT('mqccred(ChlExit)') +
REPLACE
```

Référence associée
SCYDATA
SCYEXIT
runmqccred

Authentification de connexion avec le client Java

L'authentification de connexion est une fonction de IBM MQ qui permet au gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe

fournis. Lorsque l'application est une application Java qui utilise des liaisons client, l'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

Mode compatibilité

Avant IBM MQ 8.0, le client Java pouvait envoyer un ID utilisateur et un mot de passe via le canal de connexion client au canal de connexion serveur, et les fournir à un exit de sécurité dans les zones **RemoteUserIdentifier** et **RemotePassword** de la structure MQCD. En mode compatibilité, ce comportement est conservé.

Vous pouvez utiliser ce mode en combinaison avec l'authentification de connexion et effectuer une migration à partir de tous les exits de sécurité précédemment utilisés pour effectuer le même travail.

Vous devez utiliser ADOPTCTX (YES) ou utiliser une autre méthode, par exemple une règle CHLAUTH basée sur un certificat TLS, pour définir l'utilisateur MCAUSER en cours d'exécution lorsque vous utilisez le mode de compatibilité, car dans ce mode, l'ID utilisateur côté client n'est pas envoyé au gestionnaire de files d'attente.

Le mode compatibilité des opérations peut être activé connexion par connexion ou globalement :

- Dans IBM MQ classes for Java, définissez la propriété *MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY* sur `false` dans la table de hachage des propriétés transmise au constructeur **com.ibm.mq.MQQueueManager**.
- Dans IBM MQ classes for JMS, définissez la propriété *JmsConstants.USER_AUTHENTICATION_MQCSP* sur `false`, dans la fabrique de connexions appropriée, avant de créer la connexion.
- Globalement, spécifiez la propriété système Java *-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication= N* sur la ligne de commande lors du démarrage de votre application, comme illustré dans l'exemple suivant:

```
java -Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=N application_name
```

Le mode de compatibilité est le paramètre par défaut.

Mode d'authentification MQCSP

Dans ce mode, l'ID utilisateur côté client est envoyé, ainsi que l'ID utilisateur et le mot de passe à authentifier, de sorte que vous pouvez utiliser ADOPTCTX (NO). L'ID utilisateur et le mot de passe sont disponibles pour un exit de sécurité de connexion serveur dans la structure MQCSP fournie dans la structure MQCXP.

Ce mode de fonctionnement peut être activé connexion par connexion ou globalement:

- Dans IBM MQ classes for Java, définissez la propriété *MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY* sur `true` dans la table de hachage des propriétés qui est transmise au constructeur **com.ibm.mq.MQQueueManager** .
- Dans IBM MQ classes for JMS, définissez la propriété *JmsConstants.USER_AUTHENTICATION_MQCSP* à `true`, sur la fabrique de connexions appropriée avant de créer la connexion.
- Globalement, définissez la propriété système *com.ibm.mq.cfg.jmqi.useMQCSPauthentication* sur une valeur indiquant la valeur true, par exemple, en ajoutant `-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=Y` à la ligne de commande.

Choix du mode d'authentification dans IBM MQ Explorer

IBM MQ Explorer étant une application Java , ces deux modes, le mode compatibilité et le mode d'authentification MQCSP, lui sont également applicables.

V 9.1.0 Depuis IBM MQ 9.1.0, le mode d'authentification MQCSP est le mode par défaut. Avant IBM MQ 9.1, le mode compatibilité est le mode par défaut.

Dans les panneaux où l'identification de l'utilisateur est fournie, une case à cocher permet d'activer ou de désactiver le mode de compatibilité:

- V 9.1.0 Depuis IBM MQ 9.1.0, par défaut, cette case n'est pas cochée. Pour utiliser le mode compatibilité, cochez cette case.
- Avant IBM MQ 9.1.0, par défaut, cette case est cochée. Pour utiliser l'authentification MQCSP, décochez la case.

Concepts associés

«Authentification de connexion», à la page 68

«Authentification de la connexion: modifications de l'application», à la page 73

«Authentification de connexion: référentiels d'utilisateurs», à la page 74

Pour chacun de vos gestionnaires de files d'attente, vous pouvez choisir différents types d'objet d'informations d'authentification pour l'authentification des ID utilisateur et des mots de passe.

Sécurité des messages dans IBM MQ

La sécurité des messages dans l'infrastructure IBM MQ est fournie par Advanced Message Security.

Advanced Message Security (AMS) développe les services de sécurité IBM MQ pour fournir la signature et le chiffrement des données au niveau des messages. Les services étendus garantissent que les données de message n'ont pas été modifiées entre le moment où elles sont placées à l'origine dans une file d'attente et le moment où elles sont extraites. En outre, AMS vérifie qu'un expéditeur de données de message est autorisé à placer des messages signés dans une file d'attente cible.

Concepts associés

«Advanced Message Security», à la page 580

Advanced Message Security (AMS) est un composant de IBM MQ qui offre un niveau de protection élevé pour les données sensibles transitant par le réseau IBM MQ , sans affecter les applications finales.

Planification de la sécurité

Cette collection de rubriques explique ce que vous devez prendre en compte lors de la planification de la sécurité dans un environnement IBM MQ .

Vous pouvez utiliser IBM MQ pour une grande variété d'applications sur une large gamme de plateformes. Les exigences de sécurité sont susceptibles d'être différentes pour chaque application. Pour certains, la sécurité sera une considération cruciale.

IBM MQ fournit une gamme de services de sécurité au niveau des liens, y compris la prise en charge du protocole TLS (Transport Layer Security).

Vous devez prendre en compte certains aspects de la sécurité lors de la planification de l'installation de IBM MQ:

- Multi Sous Multiplateformes, si vous ignorez ces aspects et ne faites rien, vous ne pouvez pas utiliser IBM MQ.
- z/OS Sous z/OS, le fait d'ignorer ces aspects a pour effet que vos ressources IBM MQ ne sont pas protégées. C'est-à-dire que tous les utilisateurs peuvent accéder à toutes les ressources IBM MQ et les modifier.

Droit d'administration de IBM MQ

Les administrateurs IBM MQ doivent disposer des droits suivants:

- Emettez des commandes pour administrer IBM MQ
- Utilisez laIBM MQ Explorer
- IBM i Utilisez les panneaux et les commandes d'administration IBM i .

- z/OS Utilisation des panneaux d'opérations et de contrôle sous z/OS
- z/OS Utilisez le programme utilitaire IBM MQ , CSQUTIL, sous z/OS
- z/OS Accès aux fichiers du gestionnaire de files d'attente sous z/OS

Pour plus d'informations, voir :

- ULW «Droit d'administration de IBM MQ sur UNIX, Linux, and Windows», à la page 416
- IBM i «Droit d'administration de IBM MQ sur IBM i», à la page 86
- z/OS «Droit d'administration de IBM MQ sur z/OS», à la page 87

Droits d'utilisation des objets IBM MQ

Les applications peuvent accéder aux objets IBM MQ suivants en émettant des appels MQI:

- Gestionnaires de files d'attente
- Files d'attente
- Processus
- Listes de noms
- Rubriques

Les applications peuvent également utiliser les commandes PCF (Programmable Command Format) pour accéder à ces objets IBM MQ , ainsi que pour accéder aux canaux et aux objets d'informations d'authentification. Ces objets peuvent être protégés par IBM MQ de sorte que les ID utilisateur associés aux applications aient besoin de droits d'accès.

Pour plus d'informations, voir «Autorisation pour les applications d'utiliser IBM MQ», à la page 89.

Sécurité des canaux

Les ID utilisateur associés aux agents MCA (Message Channel Agent) doivent être autorisés à accéder à diverses ressources IBM MQ . Par exemple, un agent MCA doit pouvoir se connecter à un gestionnaire de files d'attente. S'il s'agit d'un agent MCA émetteur, il doit pouvoir ouvrir la file d'attente de transmission du canal. S'il s'agit d'un agent MCA récepteur, il doit pouvoir ouvrir des files d'attente de destination. Les ID utilisateur associés aux applications qui doivent administrer les canaux, les initiateurs de canal et les programmes d'écoute doivent disposer des droits d'utilisation des commandes PCF appropriées. Cependant, la plupart des applications n'ont pas besoin d'un tel accès.

Pour plus d'informations, voir «Autorisation de canal», à la page 111.

Autres considérations

Vous devez prendre en compte les aspects suivants de la sécurité uniquement si vous utilisez certaines fonctions IBM MQ ou certaines extensions de produit de base:

- «Sécurité des clusters de gestionnaires de files d'attente», à la page 124
- «Sécurité pour la publication / abonnement IBM MQ», à la page 125
- «Sécurité de IBM MQ Internet Pass-Thru», à la page 126

Planification de l'identification et de l'authentification

Choisissez les ID utilisateur à utiliser, ainsi que la manière et les niveaux auxquels vous souhaitez appliquer les contrôles d'authentification.

Vous devez décider de la manière dont vous allez identifier les utilisateurs de vos applications IBM MQ , en gardant à l'esprit que différents systèmes d'exploitation prennent en charge des ID utilisateur de longueurs différentes. Vous pouvez utiliser des enregistrements d'authentification de canal pour effectuer

un mappage d'un ID utilisateur à un autre ou pour spécifier un ID utilisateur en fonction d'un attribut de la connexion. Les canaux IBM MQ utilisant TLS utilisent des certificats numériques comme mécanisme d'identification et d'authentification. Chaque certificat numérique possède un nom distinctif de sujet qui peut être mappé à des identités spécifiques à l'aide d'enregistrements d'authentification de canal. En outre, les certificats de l'autorité de certification dans le référentiel de clés déterminent quels certificats numériques peuvent être utilisés pour l'authentification auprès de IBM MQ. Pour plus d'informations, voir :

- «Mappage d'un gestionnaire de files d'attente éloignées à un ID utilisateur MCAUSER», à la page 399
- «Mappage d'un ID utilisateur client à un ID utilisateur MCAUSER», à la page 400
- «Mappage d'un nom distinctif SSL ou TLS à un ID utilisateur MCAUSER», à la page 401
- «Mappage d'une adresse IP à un ID utilisateur MCAUSER», à la page 403

Planification de l'authentification pour une application client

Vous pouvez appliquer des contrôles d'authentification à quatre niveaux: au niveau des communications, dans les exits de sécurité, avec des enregistrements d'authentification de canal et en termes d'identification transmise à un exit de sécurité.

Il y a quatre niveaux de sécurité à prendre en compte. Le diagramme montre un IBM MQ MQI client connecté à un serveur. La sécurité est appliquée à quatre niveaux, comme décrit dans le texte suivant. MCA est un agent MCA.

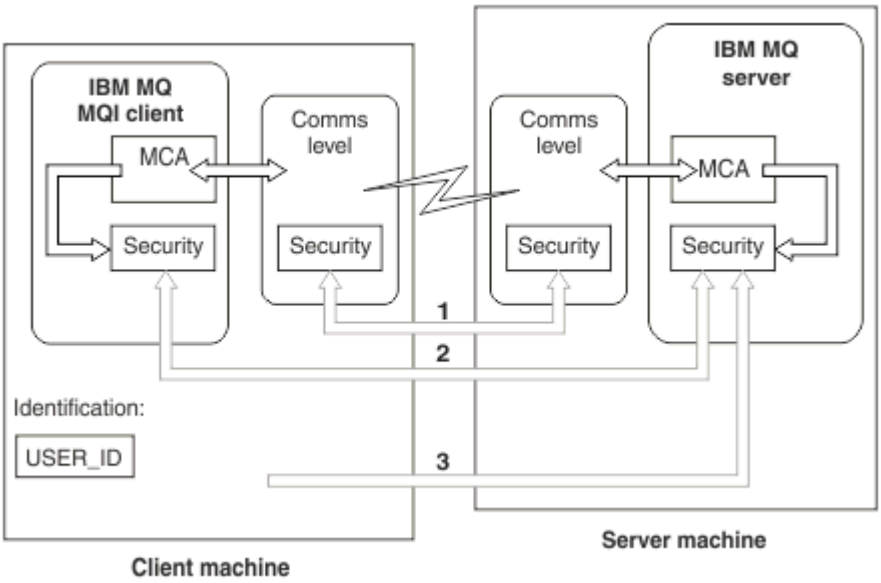


Figure 9. Sécurité dans une connexion client/serveur

1. Niveau de communication

   Voir la flèche 1. Pour implémenter la sécurité au niveau des communications, utilisez TLS. Pour plus d'informations, voir «Protocole de sécurité cryptographique TLS», à la page 15

2. Enregistrements d'authentification de canal

   Voir les flèches 2 et 3. L'authentification peut être contrôlée à l'aide de l'adresse IP ou des noms distinctifs TLS au niveau de la sécurité. Un ID utilisateur peut également être bloqué ou un ID utilisateur vérifié peut être mappé à un ID utilisateur valide. Une description complète est fournie dans «Enregistrements d'authentification de canal», à la page 50.

3. Authentification de connexion

   Voir la flèche 3. Le client envoie un ID et un mot de passe. Pour plus d'informations, voir «Authentification de connexion: Configuration», à la page 69.

4. Exits de sécurité de canal

Voir la flèche 2. Les exits de sécurité de canal pour la communication de client à serveur peuvent fonctionner de la même manière que pour la communication de serveur à serveur. Une paire d'exits indépendants du protocole peut être écrite pour permettre l'authentification mutuelle du client et du serveur. Une description complète est fournie dans Programmes d'exit de sécurité de canal.

5. Identification transmise à un exit de sécurité de canal

   Voir la flèche 3. Dans les communications client-serveur, les exits de sécurité de canal n'ont pas besoin de fonctionner en tant que paire. L'exit côté client IBM MQ peut être omis. Dans ce cas, l'ID utilisateur est placé dans le descripteur de canal (MQCD) et l'exit de sécurité côté serveur peut le modifier, si nécessaire.

   Les clients Windows envoient également des informations supplémentaires pour faciliter l'identification.

   - L'ID utilisateur transmis au serveur est l'ID utilisateur actuellement connecté sur le client.
   - ID de sécurité de l'utilisateur actuellement connecté.

   Les valeurs de l'ID utilisateur et, si elles sont disponibles, de l'ID de sécurité, peuvent être utilisées par l'exit de sécurité du serveur pour établir l'identité du IBM MQ MQI client.

Depuis IBM MQ 8.0, vous pouvez envoyer des mots de passe inclus dans la structure MQCSP.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont protégés de manière appropriée, voir «Protection par mot de passe MQCSP», à la page 30.

ID utilisateur

Lorsque vous créez des ID utilisateur pour des applications client, les ID utilisateur ne doivent pas dépasser la longueur maximale autorisée. Vous ne devez pas utiliser les ID utilisateur réservés UNKNOWN et NOBODY. Si le serveur auquel le client se connecte est un serveur IBM MQ for Windows , vous devez mettre fin à l'utilisation du signe @. La longueur autorisée des ID utilisateur dépend de la plateforme utilisée pour le serveur:

- z/OS Linux UNIX Sous z/OS et UNIX and Linux, la longueur maximale d'un ID utilisateur est de 12 caractères.
- IBM i Sous IBM i, la longueur maximale d'un ID utilisateur est de 10 caractères.
- Windows Sous Windows, si IBM MQ MQI clientet le serveur IBM MQ sont sous Windowset que le serveur a accès au domaine dans lequel l'ID utilisateur client est défini, la longueur maximale d'un ID utilisateur est de 20 caractères. Toutefois, si le serveur IBM MQ n'est pas un serveur Windows , l'ID utilisateur est tronqué à 12 caractères.
- Si vous utilisez la structure MQCSP pour transmettre des données d'identification, la longueur maximale d'un ID utilisateur est de 1024 caractères. L'ID utilisateur de la structure MQCSP ne peut pas être utilisé pour contourner la longueur maximale de l'ID utilisateur utilisée par IBM MQ pour l'autorisation. Pour plus d'informations sur la structure MQCSP, voir «Identification et authentification des utilisateurs à l'aide de la structure MQCSP», à la page 344.

Sur les systèmes UNIX and Linux , les ID utilisateur sont utilisés par défaut pour l'authentification et les groupes sont utilisés pour l'autorisation. Toutefois, vous pouvez configurer ces systèmes pour une autorisation sur les ID utilisateur. Pour plus d'informations, voir «Droits utilisateur OAM sur UNIX and Linux», à la page 362. Les systèmes Windows peuvent utiliser les deux ID utilisateur pour l'authentification et l'autorisation et les groupes pour l'autorisation.

Si vous créez des comptes de service, sans tenir compte des groupes, et que vous autorisez tous les ID utilisateur différemment, chaque utilisateur peut accéder aux informations de chaque autre utilisateur.

ID utilisateur restreints

Les ID utilisateur UNKNOWN et le groupe NOBODY ont une signification particulière pour IBM MQ. La création d'un ID utilisateur dans le système d'exploitation appelé UNKNOWN ou d'un groupe appelé NOBODY peut avoir des résultats inattendus.

ID utilisateur lors de la connexion à un serveur IBM MQ for Windows

Un serveur IBM MQ for Windows ne prend pas en charge la connexion d'un client Windows si le client s'exécute sous un ID utilisateur contenant le caractère @, par exemple, `abc@d`. Le code retour de l'appel MQCONN au niveau du client est MQRC_NOT_AUTHORIZED.

Toutefois, vous pouvez spécifier l'ID utilisateur à l'aide de deux caractères @, par exemple, `abc@@d`. Il est recommandé d'utiliser le format `id@domain` pour s'assurer que l'ID utilisateur est résolu de manière cohérente dans le domaine approprié ; par conséquent, `abc@@d@domain`.

Autorisation de planification

Planifiez les utilisateurs qui auront des droits d'administration et planifiez la manière d'autoriser les utilisateurs des applications à utiliser les objets IBM MQ de manière appropriée, y compris ceux qui se connectent à partir d'un IBM MQ MQI client.

Les personnes ou les applications doivent disposer d'un accès leur permettant d'utiliser IBM MQ. L'accès dont ils ont besoin dépend des rôles qu'ils assument et des tâches qu'ils doivent effectuer. L'autorisation dans IBM MQ peut être divisée en deux catégories principales:

- Autorisation d'effectuer des opérations d'administration
- Autorisation pour les applications d'utiliser IBM MQ

Les deux classes d'opération sont contrôlées par le même composant et un individu peut être autorisé à effectuer les deux catégories d'opération.

Les rubriques suivantes fournissent des informations supplémentaires sur des domaines d'autorisation spécifiques que vous devez prendre en compte:

Droit d'administration de IBM MQ

Les administrateurs IBM MQ doivent disposer des droits nécessaires pour exécuter diverses fonctions. Ces droits sont obtenus de différentes manières sur différentes plateformes.

Les administrateurs IBM MQ doivent disposer des droits suivants:

- Exécutez des commandes pour administrer IBM MQ.
- 

Utilisez la console IBM MQ Explorer.
- 

Utilisez les panneaux d'opérations et de contrôle sous z/OS.
- 

Utilisez le programme utilitaire IBM MQ , CSQUTIL, sous z/OS.
- 

Accédez aux fichiers du gestionnaire de files d'attente sur z/OS.

Pour plus d'informations, voir la rubrique correspondant à votre système d'exploitation.

Droits d'administration de IBM MQ sur les systèmes UNIX et Windows

Un administrateur IBM MQ est membre du groupe mqm. Ce groupe a accès à toutes les ressources IBM MQ et peut émettre des commandes de contrôle IBM MQ . Un administrateur peut accorder des droits spécifiques à un groupe d'utilisateurs.

Pour être un administrateur IBM MQ sur les systèmes UNIX et Windows , un utilisateur doit être membre du *groupe mqm*. Ce groupe est créé automatiquement lorsque vous installez IBM MQ. Pour permettre aux utilisateurs d'émettre des commandes de contrôle, vous devez les ajouter au groupe mqm. Cela inclut l'utilisateur root sous UNIX.

Les utilisateurs qui ne sont pas membres du groupe mqm peuvent se voir accorder des privilèges d'administration, mais ils ne peuvent pas émettre de commandes de contrôle IBM MQ et ils sont autorisés à exécuter uniquement les commandes auxquelles ils ont accès.

De plus, sur les systèmes Windows , les comptes SYSTEM et Administrator disposent d'un accès complet aux ressources IBM MQ .

Tous les membres du groupe mqm ont accès à toutes les ressources IBM MQ sur le système, y compris la possibilité d'administrer tout gestionnaire de files d'attente en cours d'exécution sur le système. Cet accès peut être révoqué uniquement en supprimant un utilisateur du groupe mqm. Sur les systèmes Windows , les membres du groupe Administrateurs ont également accès à toutes les ressources IBM MQ .

Les administrateurs peuvent utiliser la commande de contrôle **runmqsc** pour émettre des commandes IBM MQ Script (MQSC). Lorsque **runmqsc** est utilisé en mode indirect pour envoyer des commandes MQSC à un gestionnaire de files d'attente éloignées, chaque commande MQSC est encapsulée dans une commande Escape PCF. Les administrateurs doivent disposer des droits requis pour que les commandes MQSC soient traitées par le gestionnaire de files d'attente éloignées.

IBM MQ Explorer émet des commandes PCF pour effectuer des tâches d'administration. Les administrateurs n'ont pas besoin de droits supplémentaires pour utiliser IBM MQ Explorer afin d'administrer un gestionnaire de files d'attente sur le système local. Lorsque IBM MQ Explorer est utilisé pour administrer un gestionnaire de files d'attente sur un autre système, les administrateurs doivent disposer des droits requis pour que les commandes PCF soient traitées par le gestionnaire de files d'attente éloignées.

Pour plus d'informations sur les vérifications des droits d'accès effectuées lors du traitement des commandes PCF et MQSC, voir les rubriques suivantes:

- Pour les commandes qui fonctionnent sur les gestionnaires de files d'attente, les files d'attente, les canaux, les processus, les listes de noms et les objets d'informations d'authentification, voir «Autorisation pour les applications d'utiliser IBM MQ», à la page 89.
- Pour les commandes qui fonctionnent sur les canaux, les initiateurs de canal, les programmes d'écoute et les clusters, voir Sécurité des canaux.
- z/OS Pour les commandes MQSC traitées par le serveur de commandes sous IBM MQ for z/OS, voir «Sécurité des commandes et des ressources de commandes sous z/OS», à la page 88.

Pour plus d'informations sur les droits dont vous avez besoin pour administrer IBM MQ sur les systèmes UNIX et Windows , voir les informations connexes.

IBM i *Droit d'administration de IBM MQ sur IBM i*

Pour être un administrateur IBM MQ sous IBM i, vous devez être membre du *groupe QMQMADM*. Ce groupe possède des propriétés similaires à celles du groupe mqm sur les systèmes UNIX et Windows . En particulier, le groupe QMQMADM est créé lorsque vous installez IBM MQ for IBM iet les membres du groupe QMQMADM ont accès à toutes les ressources IBM MQ sur le système. Vous avez également accès à toutes les ressources IBM MQ si vous disposez des droits *ALLOBJ.

Les administrateurs peuvent utiliser des commandes CL pour administrer IBM MQ.L'une de ces commandes est GRTMQMAUT, qui permet d'accorder des droits à d'autres utilisateurs. Une autre commande, STRMQMMQSC, permet à un administrateur d'émettre des commandes MQSC vers un gestionnaire de files d'attente local.

Deux groupes de commandes CL sont fournis par IBM MQ for IBM i:

Groupe 1
Pour émettre une commande dans cette catégorie, un utilisateur doit être membre du groupe QMQMADM ou disposer des droits *ALLOBJ. GRTMQMAUT et STRMQMMQSC appartiennent à cette catégorie, par exemple.

Groupe 2
Pour émettre une commande dans cette catégorie, un utilisateur n'a pas besoin d'être membre du groupe QMQMADM ou de disposer des droits *ALLOBJ. Au lieu de cela, deux niveaux d'autorité sont requis:

- L'utilisateur doit disposer des droits IBM i pour utiliser la commande. Ces droits sont accordés à l'aide de la commande GRTOBJAUT.
- L'utilisateur doit disposer des droits IBM MQ pour accéder à tout objet IBM MQ associé à la commande. Ces droits sont accordés à l'aide de la commande GRTMQMAUT.

Les exemples suivants présentent les commandes de ce groupe:

- CRTMQMQ, Création d'une file d'attente MQM
- CHGMQMPRC, modification du processus MQM
- DLTMQMNL, Suppression de la liste de noms MQM
- DSPMQMAUTI, Affichage des informations d'authentification MQM
- CRTMQMCHL, création d'un canal MQM

Pour plus d'informations sur ce groupe de commandes, voir «Autorisation pour les applications d'utiliser IBM MQ», à la page 89.

Pour la liste complète des commandes du groupe 1 et du groupe 2, voir «Droits d'accès pour les objets IBM MQ sous IBM i», à la page 159

Pour plus d'informations sur les droits d'accès dont vous avez besoin pour administrer IBM MQ sur IBM i, voir Administration d' IBM i .

z/OS *Droit d'administration de IBM MQ sur z/OS*

Cette collection de rubriques décrit les différents aspects des droits dont vous avez besoin pour administrer IBM MQ for z/OS.

z/OS *Vérifications des droits d'accès sous z/OS*

IBM MQ for z/OS utilise la fonction d'autorisation système (SAF) pour acheminer les demandes de vérification des droits d'accès à un gestionnaire de sécurité externe (ESM), tel que z/OS Security Server Resource Access Control Facility (RACF). IBM MQ ne vérifie pas ses propres droits d'accès.

Il est supposé que vous utilisez RACF comme gestionnaire ESM. Si vous utilisez un gestionnaire ESM différent, vous devrez peut-être interpréter les informations fournies pour RACF d'une manière qui soit pertinente pour votre gestionnaire ESM.

Vous pouvez indiquer si vous souhaitez que les vérifications des droits d'accès soient activées ou désactivées pour chaque gestionnaire de files d'attente individuellement ou pour chaque gestionnaire de files d'attente d'un groupe de partage de files d'attente.Ce niveau de contrôle est appelé *sécurité du sous-système*. Si vous désactivez la sécurité du sous-système pour un gestionnaire de files d'attente particulier, aucune vérification des droits d'accès n'est effectuée pour ce gestionnaire de files d'attente.

Si vous activez la sécurité du sous-système pour un gestionnaire de files d'attente particulier, des vérifications des droits d'accès peuvent être effectuées à deux niveaux:

Sécurité au niveau du groupe de partage de files d'attente
Les vérifications des droits d'accès utilisent des profils RACF qui sont partagés par tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cela signifie qu'il y a moins de profils à définir et à gérer, ce qui facilite l'administration de la sécurité.

Sécurité au niveau du gestionnaire de files d'attente
Les vérifications des droits d'accès utilisent des profils RACF spécifiques au gestionnaire de files d'attente.

Vous pouvez utiliser une combinaison de sécurité au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente. Par exemple, vous pouvez organiser des profils spécifiques à un gestionnaire de files d'attente pour remplacer ceux du groupe de partage de files d'attente auquel il appartient.

La sécurité du sous-système, la sécurité au niveau du groupe de partage de files d'attente et la sécurité au niveau du gestionnaire de files d'attente sont activées ou désactivées en définissant des *profils de commutation*. Un profil de commutation est un profil RACF normal qui a une signification spéciale pour IBM MQ.

z/OS *Sécurité des commandes et des ressources de commandes sous z/OS*

La sécurité de commande est liée au droit d'émettre une commande ; le droit de ressource de commande est lié au droit d'effectuer une opération sur une ressource. Les deux sont implémentées à l'aide des classes RACF .

Les vérifications des droits d'accès sont effectuées lorsqu'un administrateur IBM MQ émet une commande MQSC. Il s'agit de la *sécurité des commandes*.

Pour implémenter la sécurité des commandes, vous devez définir certains profils RACF et accorder aux groupes et aux ID utilisateur nécessaires l'accès à ces profils aux niveaux requis. Le nom d'un profil de sécurité de commande contient le nom d'une commande MQSC.

Certaines commandes MQSC effectuent une opération sur une ressource IBM MQ , telle que la commande DEFINE QLOCAL pour créer une file d'attente locale. Lorsqu'un administrateur émet une commande MQSC, des vérifications des droits d'accès sont effectuées pour déterminer si l'opération demandée peut être effectuée sur la ressource spécifiée dans la commande. Il s'agit de la *sécurité des ressources de commande*.

Pour implémenter la sécurité des ressources de commande, vous devez définir certains profils RACF et donner aux groupes et ID utilisateur nécessaires l'accès à ces profils aux niveaux requis. Le nom d'un profil de sécurité de ressource de commande contient le nom d'une ressource IBM MQ et son type (QUEUE, PROCESS, NAMELIST, TOPIC, AUTHINFO ou CHANNEL).

La sécurité des commandes et la sécurité des ressources de commande sont indépendantes. Par exemple, lorsqu'un administrateur émet la commande:

```
DEFINE QLOCAL(MOON.EUROPA)
```

les vérifications des droits d'accès suivantes sont effectuées:

- Les contrôles de sécurité de commande vérifient que l'administrateur est autorisé à émettre la commande DEFINE QLOCAL.
- Les contrôles de sécurité des ressources de commande permettent à l'administrateur d'effectuer une opération sur la file d'attente locale appelée MOON.EUROPA.

La sécurité des commandes et des ressources de commande peut être activée ou désactivée en définissant des profils de commutateur.

z/OS *Commandes MQSC et file d'attente d'entrée des commandes système sous z/OS*

Cette rubrique explique comment le serveur de commandes traite les commandes MQSC dirigées vers la file d'attente d'entrée des commandes système sur z/OS.

La sécurité des commandes et la sécurité des ressources de commandes sont également utilisées lorsque le serveur de commandes extrait un message contenant une commande MQSC de la file d'attente d'entrée des commandes système.L'ID utilisateur utilisé pour les vérifications des droits d'accès est celui qui se trouve dans la zone *UserIdentifier* du descripteur de message du message contenant la commande MQSC. Cet ID utilisateur doit disposer des droits requis sur le gestionnaire de files d'attente dans lequel la commande est traitée. Pour plus d'informations sur la zone *UserIdentifier* et sa définition, voir Contexte de message.

Les messages contenant des commandes MQSC sont envoyés à la file d'attente d'entrée des commandes système dans les cas suivants:

- Les panneaux d'opérations et de contrôle envoient des commandes MQSC à la file d'attente d'entrée des commandes système du gestionnaire de files d'attente cible. Les commandes MQSC correspondent aux actions que vous choisissez dans les panneaux. La zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur TSO de l'administrateur.
- La fonction COMMAND du programme utilitaire IBM MQ , CSQUTIL, envoie les commandes MQSC du fichier d'entrée à la file d'attente d'entrée des commandes système du gestionnaire de files d'attente cible. Les fonctions COPY et EMPTY envoient des commandes DISPLAY QUEUE et DISPLAY STGCLASS. La zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur du travail.
- Les commandes MQSC des fichiers CSQINPX sont envoyées à la file d'attente d'entrée des commandes système du gestionnaire de files d'attente auquel l'initiateur de canal est connecté. La zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur de l'espace adresse de l'initiateur de canal.

  Aucune vérification des droits n'est effectuée lorsque des commandes MQSC sont émises à partir des fichiers CSQINP1 et CSQINP2 . Vous pouvez contrôler qui est autorisé à mettre à jour ces fichiers à l'aide de la protection des fichiers RACF .
- Dans un groupe de partage de files d'attente, un initiateur de canal peut envoyer des commandes START CHANNEL à la file d'attente d'entrée des commandes système du gestionnaire de files d'attente auquel il est connecté. Une commande est envoyée lorsqu'un canal sortant qui utilise une file d'attente de transmission partagée est démarré par déclenchement. La zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur de l'espace adresse de l'initiateur de canal.
- Une application peut envoyer des commandes MQSC à une file d'attente d'entrée de commandes système. Par défaut, la zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur associé à l'application.
- Sur les systèmes UNIX, Linux, and Windows , la commande de contrôle **runmqsc** peut être utilisée en mode indirect pour envoyer des commandes MQSC à la file d'attente d'entrée de commande système d'un gestionnaire de files d'attente sous z/OS. La zone *UserIdentifier* de chaque message est définie sur l'ID utilisateur de l'administrateur qui a émis la commande **runmqsc** .

z/OS *Accès aux fichiers du gestionnaire de files d'attente sous z/OS*

Les administrateurs IBM MQ for z/OS doivent disposer des droits d'accès aux fichiers du gestionnaire de files d'attente. Utilisez cette rubrique pour savoir quels fichiers doivent être protégés par RACF .

Ces ensembles de données sont les suivants:

- V 9.1.0 Les fichiers référencés par CSQINP1, CSQINP2et CSQINPT dans la procédure de tâche démarrée du gestionnaire de files d'attente.
- Ensembles de pages du gestionnaire de files d'attente, fichiers journaux actifs, fichiers journaux archivés et fichiers d'amorçage (BSDS)
- Fichiers référencés par CSQXLIB et CSQINPX dans la procédure de tâche démarrée de l'initiateur de canal

Vous devez protéger les fichiers afin qu'aucun utilisateur non autorisé ne puisse démarrer un gestionnaire de files d'attente ou accéder à des données de gestionnaire de files d'attente. Pour ce faire, utilisez la protection des fichiers RACF .

Autorisation pour les applications d'utiliser IBM MQ

Lorsque les applications accèdent à des objets, les ID utilisateur associés aux applications doivent disposer des droits appropriés.

Les applications peuvent accéder aux objets IBM MQ suivants en émettant des appels MQI:

- Gestionnaires de files d'attente
- Files d'attente

- Processus
- Listes de noms
- Rubriques

Les applications peuvent également utiliser des commandes PCF pour administrer des objets IBM MQ . Lorsque la commande PCF est traitée, elle utilise le contexte de droits de l'ID utilisateur qui a inséré le message PCF.

Dans ce contexte, les applications incluent celles écrites par les utilisateurs et les fournisseurs, ainsi que celles fournies avec IBM MQ for z/OS. Les applications fournies avec IBM MQ for z/OS sont les suivantes:

- Les panneaux d'opérations et de contrôle
- Le programme utilitaire IBM MQ , CSQUTIL
- L'utilitaire de gestionnaire de files d'attente de rebut, CSQUDLQH

Les applications qui utilisent IBM MQ classes for Java, IBM MQ classes for JMS, IBM MQ classes for .NETou les clients de service de messagerie pour C/C++ et .NET utilisent l'interface MQI indirectement.

Les agents MCA émettent également des appels MQI et les ID utilisateur associés aux agents MCA ont besoin de droits d'accès à ces objets IBM MQ . Pour plus d'informations sur ces ID utilisateur et sur les droits dont ils ont besoin, voir «Autorisation de canal», à la page 111.

Sous z/OS, les applications peuvent également utiliser des commandes MQSC pour accéder à ces objets IBM MQ , mais la sécurité des commandes et la sécurité des ressources de commandes permettent de vérifier les droits d'accès dans ces circonstances. z/OS Pour plus d'informations, voir «Sécurité des commandes et des ressources de commandes sous z/OS», à la page 88 et «Commandes MQSC et file d'attente d'entrée des commandes système sous z/OS», à la page 88.

Sous IBM i, un utilisateur qui émet une commande CL dans le groupe 2 peut avoir besoin de droits d'accès à un objet IBM MQ associé à la commande. Pour plus d'informations, voir «Lorsque des vérifications des droits d'accès sont effectuées», à la page 90.

Lorsque des vérifications des droits d'accès sont effectuées

Les vérifications des droits d'accès sont effectuées lorsqu'une application tente d'accéder à un gestionnaire de files d'attente, à une file d'attente, à un processus ou à une liste de noms.

Sous IBM i, des vérifications des droits d'accès peuvent également être effectuées lorsqu'un utilisateur émet une commande CL dans le groupe 2 qui accède à l'un de ces objets IBM MQ . Les vérifications sont effectuées dans les cas suivants:

Lorsqu'une application se connecte à un gestionnaire de files d'attente à l'aide d'un appel MQCONN ou MQCONNX

Le gestionnaire de files d'attente demande au système d'exploitation l'ID utilisateur associé à l'application. Le gestionnaire de files d'attente vérifie ensuite que l'ID utilisateur est autorisé à s'y connecter et conserve l'ID utilisateur pour les vérifications ultérieures.

Les utilisateurs n'ont pas besoin de se connecter à IBM MQ. IBM MQ suppose que les utilisateurs sont connectés au système d'exploitation sous-jacent et qu'ils ont été authentifiés par celui-ci.

Lorsqu'une application ouvre un objet IBM MQ à l'aide d'un appel MQOPEN ou MQPUT1

Toutes les vérifications de droits sont effectuées lorsqu'un objet est ouvert, et non lors d'un accès ultérieur. Par exemple, des vérifications des droits d'accès sont effectuées lorsqu'une application ouvre une file d'attente. Elles ne sont pas effectuées lorsque l'application insère des messages dans la file d'attente ou extrait des messages de la file d'attente.

Lorsqu'une application ouvre un objet, elle indique les types d'opération qu'elle doit effectuer sur l'objet. Par exemple, une application peut ouvrir une file d'attente pour parcourir les messages qu'elle contient, en extraire des messages, mais pas pour y placer des messages. Pour chaque type d'opération, le gestionnaire de files d'attente vérifie que l'ID utilisateur associé à l'application dispose des droits permettant d'effectuer cette opération.

Lorsqu'une application ouvre une file d'attente, les vérifications des droits d'accès sont effectuées sur l'objet nommé dans la zone `ObjectName` du descripteur d'objet. La zone `ObjectName` est utilisée sur les appels `MQOPEN` ou `MQPUT1` . Si l'objet est une file d'attente alias ou une définition de file d'attente éloignée, les vérifications des droits sont effectuées sur l'objet lui-même. Elles ne sont pas effectuées sur la file d'attente dans laquelle la file d'attente alias ou la définition de file d'attente éloignée est résolue. Cela signifie que l'utilisateur n'a pas besoin de droits pour y accéder. Limitez les droits de création de files d'attente aux utilisateurs privilégiés. Si vous ne le faites pas, les utilisateurs peuvent ignorer le contrôle d'accès normal simplement en créant un alias.

Une application peut faire explicitement référence à une file d'attente éloignée. Il définit les zones `ObjectName` et `ObjectQMgrName` du descripteur d'objet sur les noms de la file d'attente éloignée et du gestionnaire de files d'attente éloignées. Les vérifications des droits d'accès sont effectuées sur la file d'attente de transmission portant le même nom que le gestionnaire de files d'attente éloignées. Sur z/OS, une vérification est effectuée sur le profil de file d'attente RACF qui correspond au nom du gestionnaire de files d'attente éloignées. Sur Multiplateformes, une vérification est effectuée par rapport au profil RQMNAME qui correspond au nom du gestionnaire de files d'attente éloignées, si la mise en cluster est utilisée. Une application peut référencer une file d'attente de cluster de manière explicite en définissant la zone `ObjectName` dans le descripteur d'objet sur le nom de la file d'attente de cluster. Les vérifications des droits d'accès sont effectuées sur la file d'attente de transmission du cluster, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

Les droits d'accès à une file d'attente dynamique sont basés sur la file d'attente modèle dont elle est dérivée, mais ne sont pas nécessairement les mêmes ; voir la remarque 1.

L'ID utilisateur utilisé par le gestionnaire de files d'attente pour les vérifications des droits d'accès est obtenu à partir du système d'exploitation. L'ID utilisateur est obtenu lorsque l'application se connecte au gestionnaire de files d'attente. Une application dûment autorisée peut émettre un appel `MQOPEN` en spécifiant un autre ID utilisateur ; des vérifications de contrôle d'accès sont ensuite effectuées sur l'autre ID utilisateur. L'utilisation d'un autre ID utilisateur ne modifie pas l'ID utilisateur associé à l'application, mais uniquement celui utilisé pour les vérifications de contrôle d'accès.

Lorsqu'une application s'abonne à une rubrique à l'aide d'un appel `MQSUB`

Lorsqu'une application s'abonne à une rubrique, elle spécifie le type d'opération qu'elle doit effectuer. Il s'agit de créer un abonnement, de modifier un abonnement existant ou de reprendre un abonnement existant sans le modifier. Pour chaque type d'opération, le gestionnaire de files d'attente vérifie que l'ID utilisateur associé à l'application est autorisé à effectuer l'opération.

Lorsqu'une application s'abonne à une rubrique, les vérifications des droits d'accès sont effectuées sur les objets de rubrique qui se trouvent dans l'arborescence de rubriques. Les objets de rubrique se trouvent au niveau ou au-dessus du point de l'arborescence de rubriques auquel l'application s'est abonnée. Les vérifications des droits d'accès peuvent impliquer des vérifications sur plusieurs objets de rubrique. L'ID utilisateur utilisé par le gestionnaire de files d'attente pour les vérifications des droits d'accès est obtenu à partir du système d'exploitation. L'ID utilisateur est obtenu lorsque l'application se connecte au gestionnaire de files d'attente.

Le gestionnaire de files d'attente effectue des vérifications des droits d'accès sur les files d'attente d'abonné, mais pas sur les files d'attente gérées.

Lorsqu'une application supprime une file d'attente dynamique permanente à l'aide d'un appel `MQCLOSE`

Le descripteur d'objet spécifié dans l'appel `MQCLOSE` n'est pas nécessairement le même que celui renvoyé par l'appel `MQOPEN` qui a créé la file d'attente dynamique permanente. S'il est différent, le gestionnaire de files d'attente vérifie l'ID utilisateur associé à l'application qui a émis l'appel `MQCLOSE` . Il vérifie que l'ID utilisateur est autorisé à supprimer la file d'attente.

Lorsqu'une application qui ferme un abonnement pour le supprimer ne l'a pas créé, les droits appropriés sont requis pour le supprimer.

Lorsqu'une commande PCF qui agit sur un objet IBM MQ est traitée par le serveur de commandes

Cette règle inclut le cas où une commande PCF agit sur un objet d'informations d'authentification.

L'ID utilisateur utilisé pour les vérifications des droits d'accès est celui qui se trouve dans la zone `UserIdentifier` du descripteur de message de la commande PCF. Cet ID utilisateur doit disposer

des droits requis sur le gestionnaire de files d'attente dans lequel la commande est traitée. La commande MQSC équivalente encapsulée dans une commande Escape PCF est traitée de la même manière. Pour plus d'informations sur la zone `UserIdentifier` et sur la manière dont elle est définie, voir «Contexte de message», à la page 92.

IBM i **Sous IBM i, lorsqu'un utilisateur émet une commande CL dans le groupe 2 qui fonctionne sur un objet IBM MQ**

Cette règle inclut le cas où une commande CL du groupe 2 s'applique à un objet d'informations d'authentification.

Des vérifications sont effectuées pour déterminer si l'utilisateur a le droit d'opérer sur un objet IBM MQ associé à la commande. Les vérifications sont effectuées sauf si l'utilisateur est membre du groupe `QMQMADM` ou dispose des droits `*ALLOBJ` . Les droits requis dépendent du type d'opération que la commande exécute sur l'objet. Par exemple, la commande **CHGMQMQ**, Modifier la file d'attente MQM, requiert le droit de modifier les attributs de la file d'attente spécifiée par la commande. En revanche, la commande **DSPMQMQ**, Afficher la file d'attente MQM, requiert le droit d'afficher les attributs de la file d'attente spécifiée par la commande.

De nombreuses commandes s'exécutent sur plusieurs objets. Par exemple, pour exécuter la commande **DLTMQMQ**, Supprimer une file d'attente MQM, les droits suivants sont requis:

- Droit de connexion au gestionnaire de files d'attente spécifié par la commande
- Droit de suppression de la file d'attente indiqué par la commande

Certaines commandes ne fonctionnent sur aucun objet du tout. Dans ce cas, l'utilisateur n'a besoin que des droits IBM i pour exécuter l'une de ces commandes. **STRMQMLSR**, Démarrer le programme d'écoute MQM est un exemple de commande de ce type.

Droits de l'utilisateur de remplacement

Lorsqu'une application ouvre un objet ou s'abonne à une rubrique, elle peut fournir un ID utilisateur sur l'appel MQOPEN, MQPUT1ou MQSUB. Il peut demander au gestionnaire de files d'attente d'utiliser cet ID utilisateur pour les vérifications des droits d'accès au lieu de celui associé à l'application.

L'application réussit à ouvrir l'objet uniquement si les deux conditions suivantes sont remplies:

- L'ID utilisateur associé à l'application a le droit de fournir un ID utilisateur différent pour les vérifications des droits. L'application est dite disposer de *droits d'utilisateur de remplacement*.
- L'ID utilisateur fourni par l'application a le droit d'ouvrir l'objet pour les types d'opération demandés ou de s'abonner à la rubrique.

Contexte de message

Les informations de *contexte de message* permettent à l'application qui extrait un message de découvrir l'origine du message. Les informations sont conservées dans les zones du descripteur de message et les zones sont divisées en trois parties logiques

Ces parties sont les suivantes:

contexte d'identité
Ces zones contiennent des informations sur l'utilisateur de l'application qui a inséré le message dans la file d'attente.

contexte d'origine
Ces zones contiennent des informations sur l'application elle-même et sur le moment où le message a été inséré dans la file d'attente.

contexte utilisateur
Ces zones contiennent les propriétés de message que les applications peuvent utiliser pour sélectionner les messages que le gestionnaire de files d'attente doit distribuer.

Lorsqu'une application insère un message dans une file d'attente, elle peut demander au gestionnaire de files d'attente de générer les informations de contexte dans le message. Il s'agit de l'action par défaut. Il peut également indiquer que les zones de contexte ne doivent pas contenir d'informations. L'ID

utilisateur associé à une application ne nécessite aucun droit spécial pour effectuer l'une ou l'autre de ces opérations.

Une application peut définir les zones de contexte d'identité dans un message, ce qui permet au gestionnaire de files d'attente de générer le contexte d'origine ou de définir toutes les zones de contexte. Une application peut également transmettre les zones de contexte d'identité d'un message qu'elle a extrait à un message qu'elle place dans une file d'attente, ou transmettre toutes les zones de contexte. Toutefois, l'ID utilisateur associé à une application requiert des droits d'accès pour définir ou transmettre des informations de contexte. Une application indique qu'elle a l'intention de définir ou de transmettre des informations de contexte lorsqu'elle ouvre la file d'attente dans laquelle elle est sur le point d'insérer des messages et que ses droits sont vérifiés à ce stade.

Voici une brève description de chacune des zones de contexte:

contexte d'identité

UserIdentifier
ID utilisateur associé à l'application qui a inséré le message. Si le gestionnaire de files d'attente définit cette zone, elle est définie sur l'ID utilisateur obtenu à partir du système d'exploitation lorsque l'application se connecte au gestionnaire de files d'attente.

AccountingToken
Informations pouvant être utilisées pour facturer le travail effectué à la suite du message.

ApplIdentityData
Si l'ID utilisateur associé à une application est autorisé à définir les zones de contexte d'identité ou à définir toutes les zones de contexte, l'application peut définir cette zone sur n'importe quelle valeur liée à l'identité. Si le gestionnaire de files d'attente définit cette zone, elle est mise à blanc.

Contexte d'origine

PutApplType
Type de l'application qui a inséré le message ; une transaction CICS , par exemple.

PutApplName
Nom de l'application qui a inséré le message.

PutDate
Date à laquelle le message a été inséré.

PutTime
Heure à laquelle le message a été inséré.

ApplOriginData
Si l'ID utilisateur associé à une application a le droit de définir toutes les zones de contexte, l'application peut définir cette zone sur n'importe quelle valeur liée à l'origine. Si le gestionnaire de files d'attente définit cette zone, elle est mise à blanc.

Contexte utilisateur

Les valeurs suivantes sont prises en charge pour **MQINQMP** ou **MQSETMP**:

MQPD_USER _CONTEXT

La propriété est associée au contexte utilisateur.

Aucune autorisation spéciale n'est requise pour pouvoir définir une propriété associée au contexte utilisateur à l'aide de l'appel MQSETMP.

Sur un gestionnaire de files d'attente V7.0 ou ultérieure, une propriété associée au contexte utilisateur est sauvegardée comme décrit pour MQOO_SAVE_ALL_CONTEXT. Une instruction MQPUT avec MQOO_PASS_ALL_CONTEXT spécifiée entraîne la copie de la propriété du contexte sauvegardé dans le nouveau message.

MQPD_NO_CONTEXT

La propriété n'est pas associée à un contexte de message.

Une valeur non reconnue est rejetée avec MQRC_PD_ERROR. La valeur initiale de cette zone est **MQPD_NO_CONTEXT**.

Pour une description détaillée de chacune des zones de contexte, voir MQMD-Descripteur de message. Pour plus d'informations sur l'utilisation du contexte de message, voir Contexte de message.

ULW IBM i *Droits d'utilisation des objets IBM MQ sur les systèmes IBM i , UNIX, Linux, and Windows*

Le composant de service d'autorisation fourni avec IBM MQ est appelé *gestionnaire des droits d'accès aux objets* (OAM). Il fournit un contrôle d'accès via des vérifications d'authentification et d'autorisation.

Authentification.

La vérification de l'authentification effectuée par la méthode d'accès aux objets (OAM) fournie avec IBM MQ est de base et n'est effectuée que dans des circonstances spécifiques. Il n'est pas destiné à répondre aux exigences strictes attendues dans un environnement hautement sécurisé.

La méthode d'accès aux objets (OAM) effectue sa vérification d'authentification lorsqu'une application se connecte à un gestionnaire de files d'attente et les conditions suivantes sont remplies:

- Si une structure MQCSP a été fournie par l'application de connexion, et
- L'attribut *AuthenticationType* de la structure MQCSP reçoit la valeur MQCSP_AUTH_USER_ID_AND_PWD, et
- La valeur CHCKLOCL ou CHKCCLNT de l'objet AUTHINFO configuré n'est pas'NONE'

Les étapes d'authentification de la méthode d'accès aux objets (OAM) valident le mot de passe à l'aide des services du système d'exploitation, qui ont peut-être été configurés pour effectuer des vérifications supplémentaires, par exemple pour s'assurer que le nom d'utilisateur n'a pas fait trop de tentatives de test de mot de passe incorrectes.

Il est possible d'utiliser d'autres mécanismes d'authentification si vous écrivez un nouveau composant de service d'autorisation ou si vous en obtenez un auprès d'un fournisseur.

Autorisation.

Les vérifications d'autorisation sont exhaustives et visent à répondre à la plupart des exigences normales.

Les vérifications d'autorisation sont effectuées lorsqu'une application émet un appel MQI pour accéder à un gestionnaire de files d'attente, une file d'attente, un processus, une rubrique ou une liste de noms. Ils sont également exécutés à d'autres moments, par exemple lorsqu'une commande est exécutée par le serveur de commandes.

Sur les systèmes IBM i IBM i , UNIX, Linux, and Windows , le *service d'autorisation* fournit le contrôle d'accès lorsqu'une application émet un appel MQI pour accéder à un objet IBM MQ qui est un gestionnaire de files d'attente, une file d'attente, un processus, une rubrique ou une liste de noms. Cela inclut la vérification des droits d'utilisateur de remplacement et des droits de définition ou de transmission des informations de contexte.

Windows Sous Windows , la méthode d'accès aux objets (OAM) accorde aux membres du groupe Administrateurs le droit d'accéder à tous les objets IBM MQ , même lorsque le contrôle UAC est activé. En outre, sur les systèmes Windows , le compte SYSTEM dispose d'un accès complet aux ressources IBM MQ .

Le service d'autorisation permet également de vérifier les droits d'accès lorsqu'une commande PCF s'exécute sur l'un de ces objets IBM MQ ou sur un objet d'informations d'authentification. La commande MQSC équivalente encapsulée dans une commande Escape PCF est traitée de la même manière.

IBM i Sous IBM i , à moins que l'utilisateur ne soit membre du groupe QMQMADM ou qu'il ne dispose des droits *ALLOBJ, le service d'autorisation fournit également des vérifications de droits lorsqu'un utilisateur émet une commande CL dans le groupe 2 qui s'applique à l'un de ces objets IBM MQ ou à un objet d'informations d'authentification.

Le service d'autorisation est un *service installable*, ce qui signifie qu'il est implémenté par un ou plusieurs *composants de service installables*. Chaque composant est appelé à l'aide d'une interface documentée.

Cela permet aux utilisateurs et aux fournisseurs de fournir des composants pour augmenter ou remplacer ceux fournis par les produits IBM MQ .

Le composant de service d'autorisation fourni avec IBM MQ est appelé gestionnaire des droits d'accès aux objets (OAM). La méthode d'accès aux objets (OAM) est automatiquement activée pour chaque gestionnaire de files d'attente que vous créez.

La méthode d'accès aux objets (OAM) gère une liste de contrôle d'accès (ACL) pour chaque objet IBM MQ auquel elle contrôle l'accès. Sur les systèmes UNIX and Linux , seuls les ID groupe peuvent apparaître dans une liste de contrôle d'accès. Cela signifie que tous les membres d'un groupe ont les mêmes droits. Sur les systèmes 

IBM i et Windows , les ID utilisateur et les ID de groupe peuvent apparaître dans une liste de contrôle d'accès. Cela signifie que des droits peuvent être accordés à des utilisateurs et à des groupes individuels.

Une limitation de 12 caractères s'applique au groupe et à l'ID utilisateur. Les plateformes UNIX limitent généralement la longueur d'un ID utilisateur à 12 caractères. AIX et Linux ont augmenté cette limite, mais IBM MQ continue d'observer une restriction de 12 caractères sur toutes les plateformes UNIX . Si vous utilisez un ID utilisateur de plus de 12 caractères, IBM MQ le remplace par la valeur "UNKNOWN". Ne définissez pas d'ID utilisateur avec la valeur "UNKNOWN".

La méthode d'accès aux objets (OAM) peut authentifier un utilisateur et modifier les zones de contexte d'identité appropriées. Vous pouvez l'activer en spécifiant une structure de paramètres de sécurité de connexion (MQCSP) sur un appel MQCONNX. La structure est transmise à la fonction OAM Authenticate User (MQZ_AUTHENTICATE_USER), qui définit les zones de contexte d'identité appropriées. Si une connexion MQCONNX est établie à partir d'un client IBM MQ , les informations du MQCSP sont transmises au gestionnaire de files d'attente auquel le client se connecte via la connexion client et le canal de connexion serveur. Si des exits de sécurité sont définis sur ce canal, le MQCSP est transmis à chaque exit de sécurité et peut être modifié par l'exit. Les exits de sécurité peuvent également créer le MQCSP. Pour plus de détails sur l'utilisation des exits de sécurité dans ce contexte, voir Programmes d'exit de sécurité de canal.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont correctement protégés, voir Protection par mot de passe CSPIBM MQ.

Sur les systèmes UNIX, Linux et Windows , la commande de contrôle **setmqaut** accorde et révoque les droits et est utilisée pour gérer les listes de contrôle d'accès. Par exemple, la commande

```
setmqaut -m JUPITER -t queue -n MOON.EUROPA -g VOYAGER +browse +get
```

permet aux membres du groupe VOYAGER de parcourir les messages dans la file d'attente MOON.EUROPA appartenant au gestionnaire de files d'attente JUPITER. Il permet également aux membres d'extraire des messages de la file d'attente. Pour révoquer ces droits ultérieurement, entrez la commande suivante:

```
setmqaut -m JUPITER -t queue -n MOON.EUROPA -g VOYAGER -browse -get
```

La commande :

```
setmqaut -m JUPITER -t queue -n MOON.* -g VOYAGER +put
```

permet aux membres du groupe VOYAGER d'insérer des messages dans n'importe quelle file d'attente dont le nom commence par les caractères `MOON.`. MOON.* est le nom d'un profil générique.Un *profil générique* vous permet d'accorder des droits pour un ensemble d'objets à l'aide d'une seule commande **setmqaut** .

La commande de contrôle **dspmqaut** permet d'afficher les droits en cours d'un utilisateur ou d'un groupe sur un objet spécifié.La commande de contrôle **dmpmqaut** est également disponible pour afficher les droits en cours associés aux profils génériques.

IBM i Sous IBM i, un administrateur utilise la commande CL GRTMQMAUT pour accorder des droits et la commande CL RVKMQMAUT pour révoquer des droits. Les profils génériques peuvent également être utilisés.Par exemple, la commande CL:

```
GRTMQMAUT MQMNAME(JUPITER) OBJTYPE(*Q) OBJ('MOON.*') USER(VOYAGER) AUT(*PUT)
```

fournit la même fonction que l'exemple précédent d'une commande **setmqaut** ; elle permet aux membres du groupe VOYAGER de placer des messages dans n'importe quelle file d'attente dont le nom commence par les caractères `MOON.`

IBM i La commande CL DSPMQMAUT permet d'afficher les droits en cours dont dispose l'utilisateur ou le groupe pour un objet spécifié. Les commandes CL WRKMQMAUT et WRKMQMAUTD sont également disponibles pour gérer les droits en cours associés aux objets et aux profils génériques.

Si vous ne souhaitez pas de vérification des droits d'accès, par exemple, dans un environnement de test, vous pouvez désactiver la méthode d'accès aux objets (OAM).

Multi *Utilisation de PCF pour accéder aux commandes OAM*

Sur les systèmes IBM i, UNIX, Linux, and Windows , vous pouvez utiliser les commandes PCF pour accéder aux commandes d'administration OAM.

Les commandes PCF et les commandes OAM équivalentes sont les suivantes:

Tableau 8. Commandes PCF et commandes OAM équivalentes

Commande PCF	**Commande OAM**
Consulter des enregistrements de droits	dmpmqaut
Consulter les droits de l'entité	dspmqaut
Définir l'enregistrement de droits d'accès	setmqaut
Supprimer l'enregistrement de droits d'accès	setmqaut avec l'option -remove

Les commandes **setmqaut** et **dmpmqaut** sont limitées aux membres du groupe mqm. Les commandes PCF équivalentes peuvent être exécutées par des utilisateurs de n'importe quel groupe disposant des droits dsp et chg sur le gestionnaire de files d'attente.

Pour plus d'informations sur l'utilisation de ces commandes, voir Introduction to Programmable Command Formats.

z/OS *Droits d'utiliser les objets IBM MQ sur z/OS*

Sous z/OS, sept catégories de vérification des droits d'accès sont associées aux appels à l'interface MQI. Vous devez définir certains profils RACF et leur accorder un accès approprié. Utilisez le profil *RESLEVEL* pour contrôler le nombre d'ID utilisateur vérifiés.

Les sept catégories de vérification des droits d'accès associées aux appels à l'interface MQI:

Sécurité des connexions
Vérifications des droits d'accès effectuées lorsqu'une application se connecte à un gestionnaire de files d'attente

Sécurité de file d'attente
Vérifications des droits d'accès effectuées lorsqu'une application ouvre une file d'attente ou supprime une file d'attente dynamique permanente

Sécurité des processus
Vérifications des droits d'accès effectuées lorsqu'une application ouvre un objet de processus

Sécurité de la liste de noms
Vérifications des droits d'accès effectuées lorsqu'une application ouvre un objet liste de noms

sécurité de substitution
Vérifications des droits effectuées lorsqu'une application demande des droits utilisateur de remplacement lors de l'ouverture d'un objet

sécurité du contexte
Vérifications des droits d'accès effectuées lorsqu'une application ouvre une file d'attente et indique qu'elle a l'intention de définir ou de transmettre les informations de contexte dans les messages qu'elle insère dans la file d'attente

Sécurité des rubriques
Vérifications des droits d'accès effectuées lorsqu'une application ouvre une rubrique

Chaque catégorie de vérification des droits d'accès est implémentée de la même manière que la sécurité des commandes et la sécurité des ressources de commandes. Vous devez définir certains profils RACF et accorder aux groupes et ID utilisateur nécessaires l'accès à ces profils aux niveaux requis. Pour la sécurité de la file d'attente, le niveau d'accès détermine les types d'opération que l'application peut effectuer sur une file d'attente. Pour la sécurité du contexte, le niveau d'accès détermine si l'application peut:

- Transmettre toutes les zones de contexte
- Transmettre toutes les zones de contexte et définir les zones de contexte d'identité
- Transmettre et définir toutes les zones de contexte

Chaque catégorie de vérification des droits peut être activée ou désactivée en définissant des profils de commutateur.

Toutes les catégories, à l'exception de la sécurité des connexions, sont appelées collectivement *sécurité des ressources d'API*.

Par défaut, lorsqu'une vérification de sécurité de ressource d'API est effectuée suite à un appel MQI d'une application utilisant une connexion par lots, un seul ID utilisateur est vérifié. Lorsqu'une vérification est effectuée suite à un appel MQI à partir d'une application CICS ou IMS ou à partir de l'initiateur de canal, deux ID utilisateur sont vérifiés.

Toutefois, en définissant un *profil RESLEVEL*, vous pouvez contrôler si zéro, un ou deux ID utilisateur sont vérifiés. Le nombre d'ID utilisateur vérifiés est déterminé par l'ID utilisateur associé au type de connexion lorsqu'une application se connecte au gestionnaire de files d'attente et par le niveau d'accès de cet ID utilisateur au profil RESLEVEL. L'ID utilisateur associé à chaque type de connexion est:

- ID utilisateur de la tâche de connexion pour les connexions par lots
- ID utilisateur de l'espace adresse CICS pour les connexions CICS
- ID utilisateur de l'espace adresse de la région IMS pour les connexions IMS
- ID utilisateur de l'espace adresse de l'initiateur de canal pour les connexions de l'initiateur de canal

Pour plus d'informations sur les droits d'utilisation des objets IBM MQ sur z/OS, voir «Droit d'administration de IBM MQ sur z/OS», à la page 87.

Sécurité de la messagerie distante

Cette section traite des aspects de la sécurité liés à la messagerie distante.

Vous devez autoriser les utilisateurs à utiliser les fonctions IBM MQ . Il est organisé en fonction des actions à entreprendre en ce qui concerne les objets et les définitions. Exemple :

- Les gestionnaires de files d'attente peuvent être démarrés et arrêtés par des utilisateurs autorisés
- Les applications doivent se connecter au gestionnaire de files d'attente et disposer des droits d'utilisation des files d'attente
- Les canaux de transmission de messages doivent être créés et contrôlés par des utilisateurs autorisés
- Les objets sont conservés dans des bibliothèques et l'accès à ces bibliothèques peut être restreint

L'agent MCA sur un site distant doit vérifier que le message en cours de distribution provient d'un utilisateur autorisé à le faire sur ce site distant. En outre, comme les agents MCA peuvent être démarrés à

distance, il peut être nécessaire de vérifier que les processus distants qui tentent de démarrer vos agents MCA sont autorisés à le faire. Il y a quatre façons possibles pour vous de faire face à cette situation:

1. Utilisez de manière appropriée l'attribut PutAuthority de votre définition de canal RCVR, RQSTR ou CLUSRCVR pour contrôler l'utilisateur utilisé pour les vérifications d'autorisation au moment où les messages entrants sont placés dans vos files d'attente. Voir la description de la commande DEFINE CHANNEL dans le guide des commandes MQSC.
2. Implémentez des enregistrements d'authentification de canal pour rejeter les tentatives de connexion non souhaitées ou pour définir une valeur MCAUSER basée sur les éléments suivants: l'adresse IP distante, l'ID utilisateur distant, le nom distinctif du sujet TLS fourni ou le nom du gestionnaire de files d'attente distant.
3. Implémentez la vérification de la sécurité de l' *exit utilisateur* pour vous assurer que le canal de transmission de messages correspondant est autorisé. La sécurité de l'installation hébergeant le canal correspondant garantit que tous les utilisateurs sont correctement autorisés, de sorte que vous n'avez pas besoin de vérifier les messages individuels.
4. Implémentez le traitement des messages de l' *exit utilisateur* pour vous assurer que les messages individuels sont vérifiés pour l'autorisation.

IBM i *Sécurité des objets IBM MQ for IBM i*

Cette section traite des aspects de la sécurité liés à la messagerie distante.

Vous devez autoriser les utilisateurs à utiliser les fonctions IBM MQ for IBM i . Ce droit est organisé en fonction des actions à entreprendre en ce qui concerne les objets et les définitions. Exemple :

- Les gestionnaires de files d'attente peuvent être démarrés et arrêtés par des utilisateurs autorisés
- Les applications doivent se connecter au gestionnaire de files d'attente et disposer des droits permettant d'utiliser les files d'attente
- Les canaux de message doivent être créés et contrôlés par des utilisateurs autorisés

L'agent MCA sur un site distant doit vérifier que le message en cours de distribution est dérivé d'un utilisateur disposant des droits permettant d'afficher le message sur ce site distant. En outre, comme les agents MCA peuvent être démarrés à distance, il peut être nécessaire de vérifier que les processus distants qui tentent de démarrer vos agents MCA sont autorisés à le faire. Il y a quatre façons possibles pour vous de faire face à cette situation:

- Dans la définition de canal, décréter que les messages doivent contenir des droits *context* acceptables, sinon ils sont supprimés.
- Implémentez des enregistrements d'authentification de canal pour rejeter les tentatives de connexion non souhaitées ou pour définir une valeur MCAUSER basée sur l'un des éléments suivants: l'adresse IP distante, l'ID utilisateur distant, le nom distinctif TLS fourni ou le nom du gestionnaire de files d'attente distant.
- Implémentez la vérification de la sécurité de l'exit utilisateur pour vous assurer que le canal de message correspondant est autorisé. La sécurité de l'installation hébergeant le canal correspondant garantit que tous les utilisateurs sont correctement autorisés, de sorte que vous n'avez pas besoin de vérifier les messages individuels.
- Implémentez le traitement des messages d'exit utilisateur pour vous assurer que les messages individuels sont vérifiés pour l'autorisation.

Voici quelques faits sur la façon dont IBM MQ for IBM i opère la sécurité:

- Les utilisateurs sont identifiés et authentifiés par IBM i.
- Les services de gestionnaire de files d'attente appelés par les applications sont exécutés avec les droits du profil utilisateur de gestionnaire de files d'attente, mais dans le processus de l'utilisateur.
- Les services de gestionnaire de files d'attente appelés par les commandes utilisateur sont exécutés avec les droits du profil utilisateur de gestionnaire de files d'attente.

Linux UNIX *Sécurité des objets sur UNIX and Linux*

Les utilisateurs d'administration doivent faire partie du groupe mqm sur votre système (y compris root) si cet ID doit utiliser les commandes d'administration IBM MQ .

Vous devez toujours exécuter amqcrsta en tant qu'ID utilisateur "mqm".

ID utilisateur sous UNIX and Linux

Le gestionnaire de files d'attente convertit tous les identificateurs utilisateur en majuscules ou en casse mixte en minuscules. Le gestionnaire de files d'attente insère ensuite les identificateurs d'utilisateur dans la partie contextuelle d'un message ou vérifie leur autorisation. Les autorisations sont donc basées uniquement sur des identificateurs en minuscules.

Windows *Sécurité des objets sur les systèmes Windows*

Les utilisateurs d'administration doivent faire partie du groupe mqm et du groupe d'administrateurs sur les systèmes Windows si cet ID doit utiliser les commandes d'administration IBM MQ .

ID utilisateur sur les systèmes Windows

Sur les systèmes Windows , *si aucun exit de message n'est installé*, le gestionnaire de files d'attente convertit tous les identificateurs d'utilisateur en majuscules ou en minuscules. Le gestionnaire de files d'attente insère ensuite les identificateurs d'utilisateur dans la partie contextuelle d'un message ou vérifie leur autorisation. Les autorisations sont donc basées uniquement sur des identificateurs en minuscules.

ID utilisateur sur tous les systèmes

Sur les plateformes autres que Windows, les systèmes UNIX and Linux utilisent des majuscules pour les ID utilisateur dans les messages. Pour permettre à Windows, aux systèmes UNIX and Linux d'utiliser des ID utilisateur en minuscules dans les messages, l'agent MCA (Message Channel Agent) doit effectuer les conversions appropriées de caractères alphabétiques.

Pour permettre aux systèmes Windows, UNIX and Linux d'utiliser des ID utilisateur en minuscules dans les messages, les conversions suivantes sont effectuées par l'agent MCA (Message Channel Agent) sur ces plateformes:

A la fin de l'envoi
Les caractères alphabétiques de tous les ID utilisateur sont convertis en caractères majuscules si aucun exit de message n'est installé.

A l'extrémité réceptrice
Les caractères alphabétiques de tous les ID utilisateur sont convertis en minuscules si aucun exit de message n'est installé.

Les conversions automatiques ne sont pas effectuées si vous fournissez un exit de message sur UNIX, Linux, and Windows pour une autre raison.

Utilisation d'un service d'autorisation personnalisé

IBM MQ fournit un service d'autorisation installable. Vous pouvez choisir d'installer un autre service.

Le composant de service d'autorisation fourni avec IBM MQ est appelé Object Authority Manager (OAM). Si la méthode d'accès aux objets (OAM) ne fournit pas les fonctions d'autorisation dont vous avez besoin, vous pouvez écrire votre propre composant de service d'autorisation. Les fonctions de service installables qui doivent être implémentées par un composant de service d'autorisation sont décrites dans Informations de référence de l'interface des services installables.

Contrôle d'accès pour les clients

Le contrôle d'accès est basé sur les ID utilisateur. Il peut y avoir de nombreux ID utilisateur à administrer, et les ID utilisateur peuvent être dans des formats différents. Vous pouvez définir la propriété de canal de connexion serveur MCAUSER sur une valeur d'ID utilisateur spéciale à utiliser par les clients.

Le contrôle d'accès dans IBM MQ est basé sur les ID utilisateur. L'ID utilisateur du processus qui effectue des appels MQI est normalement utilisé. Pour les clients MQ MQI, l'agent MCA de connexion serveur effectue des appels MQI pour le compte des clients MQ MQI. Vous pouvez sélectionner un autre ID utilisateur pour l'agent MCA de connexion serveur à utiliser pour effectuer des appels MQI. L'ID utilisateur alternatif peut être associé au poste de travail client ou à tout élément que vous choisissez pour organiser et contrôler l'accès des clients. L'ID utilisateur doit disposer des droits nécessaires sur le serveur pour émettre des appels MQI. Il est préférable de choisir un autre ID utilisateur que d'autoriser les clients à effectuer des appels MQI avec les droits de l'agent MCA de connexion serveur.

Tableau 9. ID utilisateur utilisé par un canal de connexion serveur

ID utilisateur	En cas d'utilisation
ID utilisateur défini par un exit de sécurité	Utilisé sauf s'il est bloqué par une règle **CHLAUTH TYPE(BLOCKUSER)** . Pour plus d'informations, voir la section suivante, «Définition de l'ID utilisateur dans un exit de sécurité», à la page 101 .
ID utilisateur défini par une règle CHLAUTH	Utilisé sauf si remplacé par un exit de sécurité. Pour plus d'informations, voir Enregistrements d'authentification de canal .
ID utilisateur défini dans l'attribut **MCAUSER** de la définition de canal SVRCONN	Utilisé sauf s'il est remplacé par un exit de sécurité ou une règle CHLAUTH.
ID utilisateur transmis à partir de la machine client	Utilisé lorsqu'aucun ID utilisateur n'est défini par d'autres moyens.
ID utilisateur ayant démarré le canal de connexion serveur	Utilisé lorsqu'aucun ID utilisateur n'est défini par d'autres moyens et qu'aucun ID utilisateur client n'est transmis. Pour plus d'informations, voir la section suivante, «ID utilisateur qui exécute le programme de canal», à la page 101 .

Etant donné que l'agent MCA de connexion serveur effectue des appels MQI pour le compte d'utilisateurs distants, il est important de prendre en compte les implications de sécurité de l'agent MCA de connexion serveur qui émet des appels MQI pour le compte de clients distants et de savoir comment administrer l'accès d'un nombre potentiellement élevé d'utilisateurs.

- L'une des approches consiste pour l'agent MCA de connexion serveur à émettre des appels MQI avec ses propres droits d'accès. Mais attention, il est normalement indésirable pour l'agent MCA de connexion serveur, avec ses puissantes fonctions d'accès, d'émettre des appels MQI pour le compte des utilisateurs client.
- Une autre approche consiste à utiliser l'ID utilisateur qui provient du client. L'agent MCA de connexion serveur peut émettre des appels MQI à l'aide des fonctions d'accès de l'ID utilisateur client. Cette approche pose un certain nombre de questions à prendre en considération:
  1. Il existe différents formats pour l'ID utilisateur sur différentes plateformes. Cela provoque parfois des problèmes si le format de l'ID utilisateur sur le client diffère des formats acceptables sur le serveur.
  2. Il existe potentiellement de nombreux clients, avec des ID utilisateur différents et qui changent. Les ID doivent être définis et gérés sur le serveur.
  3. L'ID utilisateur est-il digne de confiance? Tout ID utilisateur peut être transmis à partir d'un client, pas nécessairement l'ID de l'utilisateur connecté. Par exemple, le client peut transmettre un ID avec des droits `mqm` complets qui ont été définis intentionnellement uniquement sur le serveur pour des raisons de sécurité.
- L'approche préférée consiste à définir des jetons d'identification client sur le serveur, et donc à limiter les capacités des applications connectées au client. Cette opération est généralement effectuée en définissant la propriété de canal de connexion serveur MCAUSER sur une valeur d'ID utilisateur spéciale

à utiliser par les clients et en définissant quelques ID à utiliser par les clients ayant un niveau d'autorisation différent sur le serveur.

Définition de l'ID utilisateur dans un exit de sécurité

Pour IBM MQ MQI clients, le processus qui émet les appels MQI est l'agent MCA de connexion serveur. L'ID utilisateur utilisé par l'agent MCA de connexion serveur est contenu dans les zones `MCAUserIdentifier` ou `LongMCAUserIdentifier` du MQCD. Le contenu de ces zones est défini par:

- Toutes les valeurs définies par les exits de sécurité
- ID utilisateur du client
- MCAUSER (dans la définition de canal de connexion serveur)

L'exit de sécurité peut remplacer les valeurs qui lui sont visibles lorsqu'il est appelé.

- Si l'attribut MCAUSER du canal de connexion serveur est défini sur une valeur non vide, la valeur MCAUSER est utilisée.
- Si l'attribut MCAUSER du canal de connexion serveur est vide, l'ID utilisateur reçu du client est utilisé.
- Si l'attribut MCAUSER du canal de connexion serveur est vide et qu'aucun ID utilisateur n'est reçu du client, l'ID utilisateur qui a démarré le canal de connexion serveur est utilisé.

Le client IBM MQ ne transite pas l'ID utilisateur vérifié vers le serveur lorsqu'un exit de sécurité côté client est en cours d'utilisation.

ID utilisateur qui exécute le programme de canal

Lorsque les zones d'ID utilisateur sont dérivées de l'ID utilisateur qui a démarré le canal de connexion serveur, la valeur suivante est utilisée:

- z/OS Pour z/OS, ID utilisateur affecté à la tâche démarrée de l'initiateur de canal par la table des procédures démarrées z/OS .
- Pour TCP/IP (non z/OS), l'ID utilisateur de l'entrée `inetd.conf` ou l'ID utilisateur qui a démarré le programme d'écoute.
- Pour SNA (non z/OS), l'ID utilisateur de l'entrée SNA Server ou (s'il n'y en a pas) la demande de connexion entrante, ou l'ID utilisateur qui a démarré le programme d'écoute.
- Pour NetBIOS ou SPX, l'ID utilisateur qui a démarré le programme d'écoute.

S'il existe des définitions de canal de connexion serveur dont l'attribut MCAUSER est à blanc, les clients peuvent utiliser cette définition de canal pour se connecter au gestionnaire de files d'attente avec des droits d'accès déterminés par l'ID utilisateur fourni par le client. Il peut s'agir d'un risque de sécurité si le système sur lequel s'exécute le gestionnaire de files d'attente autorise des connexions réseau non autorisées. Le canal de connexion serveur par défaut IBM MQ (SYSTEM.DEF.SVRCONN) a l'attribut MCAUSER à blanc. Pour éviter les accès non autorisés, mettez à jour l'attribut MCAUSER de la définition par défaut avec un ID utilisateur qui n'a pas accès aux objets IBM MQ MQ .

Casse des ID utilisateur

Lorsque vous définissez un canal avec `runmqsc`, l'attribut MCAUSER est mis en majuscules sauf si l'ID utilisateur est placé entre apostrophes.

ULW Pour les serveurs sous UNIX, Linux, and Windows, le contenu de la zone `MCAUserIdentifier` reçue du client est remplacé par des minuscules.

IBM i Pour les serveurs sous IBM i, le contenu de la zone `LongMCAUserIdentifier` reçue du client est mis en majuscules.

Linux UNIX Pour les serveurs sur les systèmes UNIX and Linux , le contenu de la zone `LongMCAUserIdentifier` reçue du client est remplacé par des minuscules.

Par défaut, l'ID utilisateur transmis lorsqu'une application de liaison IBM MQ JMS est utilisée correspond à l'ID utilisateur de la machine virtuelle Java sur laquelle l'application est exécutée.

Il est également possible de transmettre un ID utilisateur via la méthode `createQueueConnection` .

Planification de la confidentialité

Planifiez le maintien de la confidentialité de vos données.

Vous pouvez implémenter la confidentialité au niveau de l'application ou au niveau du lien. Vous pouvez choisir d'utiliser TLS, auquel cas vous devez planifier votre utilisation des certificats numériques. Vous pouvez également utiliser des programmes d'exit de canal si les fonctions standard ne répondent pas à vos besoins.

Concepts associés

«Comparatif de la sécurité au niveau des liaisons et de la sécurité au niveau de l'application», à la page 102

Cette rubrique contient des informations sur divers aspects de la sécurité au niveau des liens et de la sécurité au niveau des applications, et compare les deux niveaux de sécurité.

«Programmes d'exit de canal», à la page 107

Les *programmes d'exit de canal* sont des programmes appelés à des endroits définis dans la séquence de traitement d'un agent MCA. Les utilisateurs et les fournisseurs peuvent écrire leurs propres programmes d'exit de canal. Certains sont fournis par IBM.

«Protection des canaux avec SSL/TLS», à la page 114

La prise en charge de TLS dans IBM MQ utilise l'objet d'informations d'authentification du gestionnaire de files d'attente et diverses commandes MQSC. Vous devez également tenir compte de votre utilisation des certificats numériques.

Comparatif de la sécurité au niveau des liaisons et de la sécurité au niveau de l'application

Cette rubrique contient des informations sur divers aspects de la sécurité au niveau des liens et de la sécurité au niveau des applications, et compare les deux niveaux de sécurité.

La sécurité au niveau des liens et des applications est illustrée dans Figure 10, à la page 102.

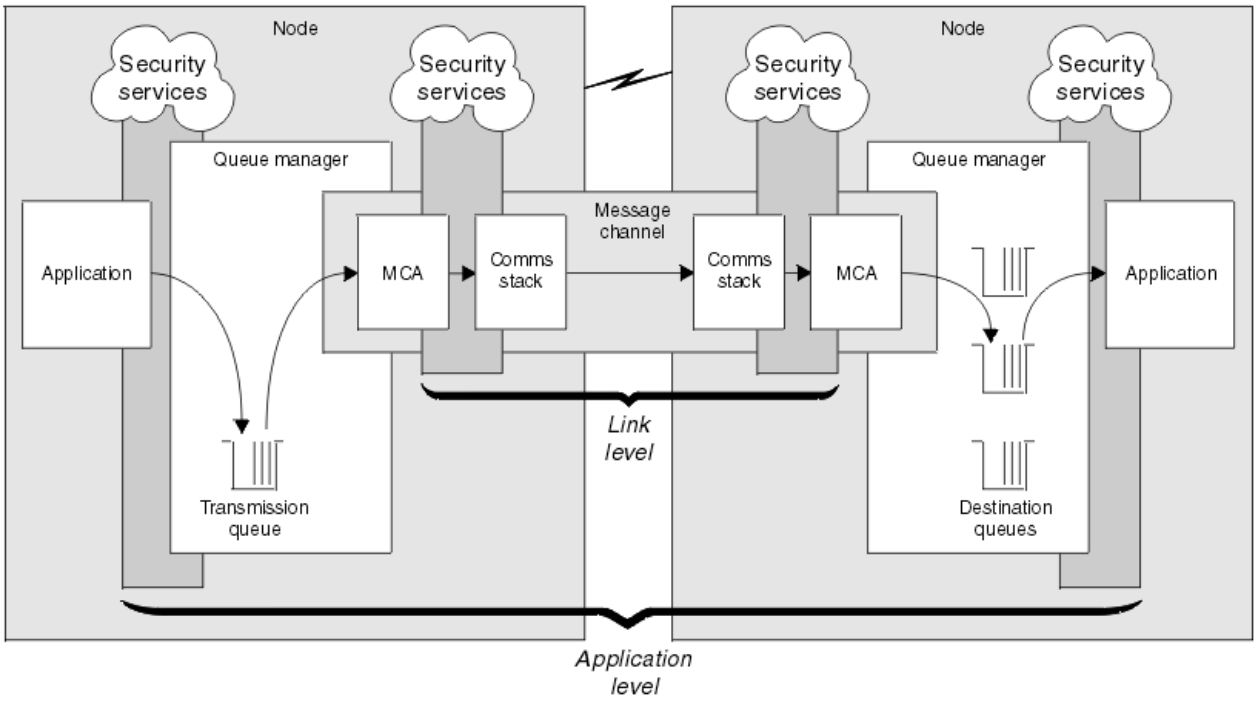


Figure 10. Sécurité au niveau des liens et sécurité au niveau des applications

Protection des messages dans les files d'attente

La sécurité au niveau des liaisons peut protéger les messages lorsqu'ils sont transférés d'un gestionnaire de files d'attente à un autre. Il est particulièrement important lorsque les messages sont transmis sur un réseau non sécurisé. Toutefois, il ne peut pas protéger les messages lorsqu'ils sont stockés dans des files d'attente d'un gestionnaire de files d'attente source, d'un gestionnaire de files d'attente de destination ou d'un gestionnaire de files d'attente intermédiaire.

V 9.1.4 z/OS Le chiffrement des fichiers z/OS peut fournir une certaine protection des messages stockés dans les files d'attente, mais uniquement pour les données au repos sur un gestionnaire de files d'attente local. Voir la section Confidentiality for data at rest on IBM MQ for z/OS with data set encryption pour plus d'informations.

La sécurité au niveau de l'application, par comparaison, peut protéger les messages lorsqu'ils sont stockés dans des files d'attente et s'applique même lorsque la mise en file d'attente répartie n'est pas utilisée. Il s'agit de la différence majeure entre la sécurité au niveau de la liaison et la sécurité au niveau de l'application. Elle est illustrée dans la Figure 10, à la page 102.

Les gestionnaires de files d'attente ne s'exécutent pas dans des environnements contrôlés et sécurisés

Si un gestionnaire de files d'attente s'exécute dans un environnement contrôlé et sécurisé, les mécanismes de contrôle d'accès fournis par IBM MQ peuvent être considérés comme suffisants pour protéger les messages stockés dans ses files d'attente. Cela est particulièrement vrai si seule la mise en file d'attente locale est impliquée et que les messages ne quittent jamais le gestionnaire de files d'attente. Dans ce cas, la sécurité au niveau de l'application peut être considérée comme inutile.

La sécurité au niveau de l'application peut également être considérée comme inutile si des messages sont transférés vers un autre gestionnaire de files d'attente qui s'exécute également dans un environnement contrôlé et sécurisé ou s'ils sont reçus d'un tel gestionnaire de files d'attente. La sécurité au niveau de l'application est d'autant plus nécessaire lorsque des messages sont transférés vers ou reçus d'un gestionnaire de files d'attente qui ne s'exécute pas dans un environnement contrôlé et sécurisé.

Différences de coût

La sécurité au niveau de l'application peut coûter plus cher que la sécurité au niveau de la liaison en termes d'administration et de performances.

Le coût de l'administration est probablement plus élevé car il y a potentiellement plus de contraintes à configurer et à gérer. Par exemple, vous pouvez être amené à vous assurer qu'un utilisateur particulier n'envoie que certains types de message et qu'il n'envoie des messages qu'à certaines destinations. A l'inverse, il peut être nécessaire de s'assurer qu'un utilisateur particulier ne reçoit que certains types de message et qu'il ne reçoit que des messages provenant de certaines sources. Au lieu de gérer les services de sécurité au niveau des liens sur un canal de messages unique, vous devrez peut-être configurer et gérer des règles pour chaque paire d'utilisateurs qui échangent des messages sur ce canal.

Il peut y avoir un impact sur les performances si les services de sécurité sont appelés à chaque fois qu'une application insère ou reçoit un message.

Les organisations ont tendance à prendre en compte d'abord la sécurité au niveau des liens car elle peut être plus facile à mettre en oeuvre. Ils prennent en compte la sécurité au niveau de l'application s'ils découvrent que la sécurité au niveau des liens ne répond pas à toutes leurs exigences.

Disponibilité des composants

Généralement, dans un environnement distribué, un service de sécurité requiert un composant sur au moins deux systèmes. Par exemple, un message peut être chiffré sur un système et déchiffré sur un autre. Cela s'applique à la sécurité au niveau de la liaison et à la sécurité au niveau de l'application.

Dans un environnement hétérogène, avec des plateformes différentes en cours d'utilisation, chacune avec des niveaux de fonction de sécurité différents, les composants requis d'un service de sécurité peuvent ne pas être disponibles pour chaque plateforme sur laquelle ils sont nécessaires et sous une forme facile à utiliser. Il s'agit probablement d'un problème plus important pour la sécurité au niveau de l'application que pour la sécurité au niveau de la liaison, en particulier si vous prévoyez de fournir votre propre sécurité au niveau de l'application en achetant des composants à partir de diverses sources.

Messages dans une file d'attente de rebut

Si un message est protégé par la sécurité au niveau de l'application, il peut y avoir un problème si, pour une raison quelconque, le message n'atteint pas sa destination et est placé dans une file d'attente de messages non livrés. Si vous ne savez pas comment traiter le message à partir des informations du descripteur de message et de l'en-tête de la lettre morte, vous devrez peut-être inspecter le contenu des données d'application. Vous ne pouvez pas effectuer cette opération si les données de l'application sont chiffrées et que seul le destinataire prévu peut les déchiffrer.

Ce que la sécurité au niveau de l'application ne peut pas faire

La sécurité au niveau de l'application n'est pas une solution complète. Même si vous implémentez la sécurité au niveau de l'application, vous pouvez tout de même avoir besoin de certains services de sécurité au niveau de la liaison. Exemple :

- Lorsqu'un canal démarre, l'authentification mutuelle des deux agents MCA peut toujours être requise. Cette opération ne peut être effectuée que par un service de sécurité au niveau de la liaison.
- La sécurité au niveau de l'application ne peut pas protéger l'en-tête de la file d'attente de transmission, MQXQH, qui inclut le descripteur de message imbriqué. Il ne peut pas non plus protéger les données dans les flux de protocole de canal IBM MQ autres que les données de message. Seule la sécurité au niveau de la liaison peut fournir cette protection.
- Si les services de sécurité au niveau de l'application sont appelés à l'extrémité serveur d'un canal MQI, les services ne peuvent pas protéger les paramètres des appels MQI envoyés via le canal. En particulier, les données d'application d'un appel MQPUT, MQPUT1ou MQGET ne sont pas protégées. Seule la sécurité au niveau des liens peut fournir la protection dans ce cas.

sécurité au niveau des liaisons

La *sécurité de niveau liaison* fait référence aux services de sécurité qui sont appelés, directement ou indirectement, par un agent MCA, le sous-système de communication ou une combinaison des deux.

La sécurité au niveau des liens est illustrée dans la Figure 10, à la page 102.

Voici quelques exemples de services de sécurité au niveau des liens:

- L'agent MCA à chaque extrémité d'un canal de transmission de messages peut authentifier son partenaire. Cette opération est effectuée lorsque le canal démarre et qu'une connexion de communication a été établie, mais avant que les messages ne commencent à circuler. Si l'authentification échoue à l'une des extrémités, le canal est fermé et aucun message n'est transféré. Il s'agit d'un exemple de service d'identification et d'authentification.
- Un message peut être chiffré à l'extrémité émettrice d'un canal et déchiffré à l'extrémité réceptrice. Il s'agit d'un exemple de service de confidentialité.
- Un message peut être vérifié à l'extrémité réceptrice d'un canal pour déterminer si son contenu a été volontairement modifié lors de sa transmission sur le réseau. Voici un exemple de service d'intégrité des données.

Sécurité au niveau de la liaison fournie par IBM MQ

Le principal moyen de mise à disposition de la confidentialité et de l'intégrité des données dans IBM MQ consiste à utiliser TLS. Pour plus d'informations sur l'utilisation de TLS dans IBM MQ, voir «Protocoles de sécurité TLS dans IBM MQ», à la page 24. Pour l'authentification, IBM MQ fournit la fonction permettant d'utiliser les enregistrements d'authentification de canal. Les enregistrements d'authentification de canal offrent un contrôle précis de l'accès accordé aux systèmes de connexion, au niveau des canaux

individuels ou des groupes de canaux. Pour plus d'informations, voir «Enregistrements d'authentification de canal», à la page 50.

Mise à disposition de votre propre sécurité de niveau de liaison
Vous pouvez fournir vos propres services de sécurité au niveau des liens. L'écriture de vos propres programmes d'exit de canal est le principal moyen de fournir vos propres services de sécurité de niveau de liaison.

Les programmes d'exit de canal sont introduits dans «Programmes d'exit de canal», à la page 107. La même rubrique décrit également le programme d'exit de canal fourni avec IBM MQ for Windows (programme d'exit de canal SSPI). Ce programme d'exit de canal est fourni au format source afin que vous puissiez modifier le code source en fonction de vos besoins. Si ce programme d'exit de canal ou les programmes d'exit de canal disponibles auprès d'autres fournisseurs ne répondent pas à vos besoins, vous pouvez concevoir et écrire vos propres programmes. Cette rubrique explique comment les programmes d'exit de canal peuvent fournir des services de sécurité. Pour plus d'informations sur l'écriture d'un programme d'exit de canal, voir Ecriture de programmes d'exit de canal.

Sécurité au niveau de la liaison à l'aide d'un exit de sécurité
Les exits de sécurité fonctionnent normalement par paires, une à chaque extrémité d'un canal. Ils sont appelés immédiatement après la fin de la négociation de données initiale au démarrage du canal.

Les exits de sécurité peuvent être utilisés pour fournir l'identification et l'authentification, le contrôle d'accès et la confidentialité.

Sécurité au niveau de la liaison à l'aide d'un exit de message
Un exit de message ne peut être utilisé que sur un canal de transmission de messages et non sur un canal MQI. Il a accès à l'en-tête de la file d'attente de transmission, MQXQH, qui inclut le descripteur de message imbriqué, et aux données d'application d'un message. Il peut modifier le contenu du message et modifier sa longueur.

Un exit de message peut être utilisé à n'importe quelle fin qui nécessite l'accès à l'ensemble du message plutôt qu'à une partie de celui-ci.

Les exits de message peuvent être utilisés pour fournir l'identification et l'authentification, le contrôle d'accès, la confidentialité, l'intégrité des données et la non-répudiation, et pour des raisons autres que la sécurité.

Sécurité au niveau de la liaison à l'aide des exits d'envoi et de réception
Les exits d'envoi et de réception peuvent être utilisés sur les canaux de message et MQI. Ils sont appelés pour tous les types de données qui circulent sur un canal et pour les flux dans les deux sens.

Les exits d'émission et de réception ont accès à chaque segment de transmission. Ils peuvent modifier son contenu et sa longueur.

Sur un canal de transmission, si un MCA a besoin de fractionner un message et de l'envoyer dans plus d'un segment de transmission, une sortie d'émission est appelée pour chaque segment de transmission contenant une partie du message et, à la réception, une sortie de réception est appelée pour chaque segment de transmission. Il en est de même sur un canal MQI si les paramètres d'entrée ou de sortie d'un appel MQI sont trop grands pour être envoyés dans un segment de transmission unique.

Sur un canal MQI, l'octet 10 d'un segment de transmission identifie l'appel MQI et indique si le segment de transmission contient les paramètres d'entrée ou de sortie de l'appel. Les exits d'envoi et de réception peuvent examiner cet octet pour déterminer si l'appel MQI contient des données d'application qui peuvent avoir besoin d'être protégées.

Lorsqu'un exit d'émission est appelé pour la première fois, pour acquérir et initialiser les ressources dont il a besoin, il peut demander à l'agent MCA de réserver une quantité d'espace spécifiée dans la mémoire tampon qui contient un segment de transmission. Lorsqu'il est appelé ultérieurement pour traiter un segment de transmission, il peut utiliser cet espace pour ajouter une clé chiffrée ou une signature numérique, par example. L'exit de réception correspondent à l'autre extrémité du canal peut supprimer les données ajoutées par l'exit d'émission et les utiliser pour traiter le segment de transmission.

Les sorties d'émission et de réception sont mieux adaptées à des fins dans lesquelles elles n'ont pas besoin de comprendre la structure des données qu'elles traitent et peuvent donc traiter chaque segment de transmission comme un objet binaire.

Les exits d'envoi et de réception peuvent être utilisés pour assurer la confidentialité et l'intégrité des données, ainsi que pour des utilisations autres que la sécurité.

Tâches associées

Identification de l'appel API dans un programme d'exit d'envoi ou de réception

sécurité au niveau de l'application

La *sécurité au niveau de l'application* fait référence aux services de sécurité appelés à l'interface entre une application et un gestionnaire de files d'attente auquel elle est connectée.

Ces services sont appelés lorsque l'application émet des appels MQI au gestionnaire de files d'attente. Les services peuvent être appelés, directement ou indirectement, par l'application, le gestionnaire de files d'attente, un autre produit prenant en charge IBM MQou une combinaison de ces deux éléments. La sécurité au niveau de l'application est illustrée dans la Figure 10, à la page 102.

La sécurité au niveau de l'application est également appelée *sécurité de bout en bout* ou *sécurité au niveau des messages*.

Voici quelques exemples de services de sécurité au niveau de l'application:

- Lorsqu'une application place un message dans une file d'attente, le descripteur de message contient un ID utilisateur associé à l'application. Toutefois, il n'existe aucune donnée, telle qu'un mot de passe chiffré, qui peut être utilisée pour authentifier l'ID utilisateur. Un service de sécurité peut ajouter ces données. Lorsque le message est finalement extrait par l'application de réception, un autre composant du service peut authentifier l'ID utilisateur à l'aide des données qui ont été transmises avec le message. Il s'agit d'un exemple de service d'identification et d'authentification.
- Un message peut être chiffré lorsqu'il est placé dans une file d'attente par une application et déchiffré lorsqu'il est extrait par l'application réceptrice.Il s'agit d'un exemple de service de confidentialité.
- Un message peut être vérifié lorsqu'il est extrait par l'application de réception. Cette vérification détermine si son contenu a été délibérément modifié depuis sa première mise en file d'attente par l'application émettrice.Voici un exemple de service d'intégrité des données.

Planification de Advanced Message Security

Advanced Message Security (AMS) est un composant de IBM MQ qui offre un niveau de protection élevé pour les données sensibles transitant par le réseau IBM MQ , sans affecter les applications finales.

Si vous déplacez des informations très sensibles ou précieuses, en particulier des informations confidentielles ou liées au paiement, telles que les dossiers des patients ou les détails de la carte de crédit, vous devez accorder une attention particulière à la sécurité de l'information. S'assurer que les informations qui circulent dans l'entreprise conservent leur intégrité et sont protégées contre tout accès non autorisé constitue un défi et une responsabilité permanents. Vous êtes également susceptible d'être tenu de respecter les règles de sécurité, au risque de sanctions en cas de non-conformité.

Vous pouvez développer vos propres extensions de sécurité dans IBM MQ. Cependant, de telles solutions nécessitent des compétences spécialisées et peuvent être compliquées et coûteuses à maintenir. Advanced Message Security vous aide à relever ces défis lorsque vous déplacez des informations dans l'entreprise entre pratiquement tous les types de système informatique commercial.

Advanced Message Security étend les fonctions de sécurité de IBM MQ comme suit:

- Il fournit une protection des données de bout en bout au niveau de l'application pour votre infrastructure de messagerie point à point, à l'aide du chiffrement ou de la signature numérique des messages.
- Il fournit une sécurité complète sans écrire de code de sécurité complexe ni modifier ou recompiler les applications existantes.
- Il utilise la technologie PKI (Public Key Infrastructure) pour fournir des services d'authentification, d'autorisation, de confidentialité et d'intégrité des données pour les messages.

- Il fournit l'administration des règles de sécurité pour les grands systèmes et les serveurs distribués.
- Il prend en charge les serveurs et les clients IBM MQ .
- Il s'intègre à Managed File Transfer pour fournir une solution de messagerie sécurisée de bout en bout.

Pour plus d'informations, voir «Advanced Message Security», à la page 580.

Mise à disposition de votre propre sécurité au niveau de l'application
Vous pouvez fournir vos propres services de sécurité au niveau de l'application. Pour vous aider à implémenter la sécurité au niveau de l'application, IBM MQ fournit deux exits, l'exit d'API et l'exit de croisement d'API.

L'exit d'API et l'exit de croisement d'API peuvent fournir des services d'identification et d'authentification, de contrôle d'accès, de confidentialité, d'intégrité des données et de non-répudiation, ainsi que d'autres fonctions non liées à la sécurité.

Si l'exit d'API ou l'exit de croisement d'API n'est pas pris en charge dans votre environnement système, vous pouvez envisager d'autres moyens de fournir votre propre sécurité au niveau de l'application. L'une des méthodes consiste à développer une API de niveau supérieur qui encapsule l'interface MQI. Les programmeurs utilisent ensuite cette API, à la place de l'interface MQI, pour écrire des applications IBM MQ .

Les raisons les plus courantes de l'utilisation d'une API de niveau supérieur sont les suivantes:

- Pour masquer les fonctions plus avancées de l'interface MQI aux programmeurs.
- Pour appliquer des normes dans l'utilisation de l'interface MQI.
- Pour ajouter une fonction à l'interface MQI. Cette fonction supplémentaire peut être des services de sécurité.

Certains produits fournisseurs utilisent cette technique pour fournir une sécurité au niveau de l'application pour IBM MQ.

Si vous prévoyez de fournir des services de sécurité de cette manière, notez ce qui suit concernant la conversion des données:

- Si un jeton de sécurité, tel qu'une signature numérique, a été ajouté aux données d'application dans un message, tout code effectuant une conversion de données doit être conscient de la présence de ce jeton.
- Un jeton de sécurité peut avoir été dérivé d'une image binaire des données d'application. Par conséquent, toute vérification du jeton doit être effectuée avant la conversion des données.
- Si les données d'application d'un message ont été chiffrées, elles doivent être déchiffrées avant la conversion des données.

Programmes d'exit de canal

Les *programmes d'exit de canal* sont des programmes appelés à des endroits définis dans la séquence de traitement d'un agent MCA. Les utilisateurs et les fournisseurs peuvent écrire leurs propres programmes d'exit de canal. Certains sont fournis par IBM.

Il existe plusieurs types de programme d'exit de canal, mais seuls quatre ont un rôle à jouer pour assurer la sécurité au niveau des liens:

- Exit de sécurité
- Exit de message
- Exit d'émission
- Exit de réception

Ces quatre types de programme d'exit de canal sont illustrés dans la Figure 11, à la page 108 et sont décrits dans les rubriques suivantes.

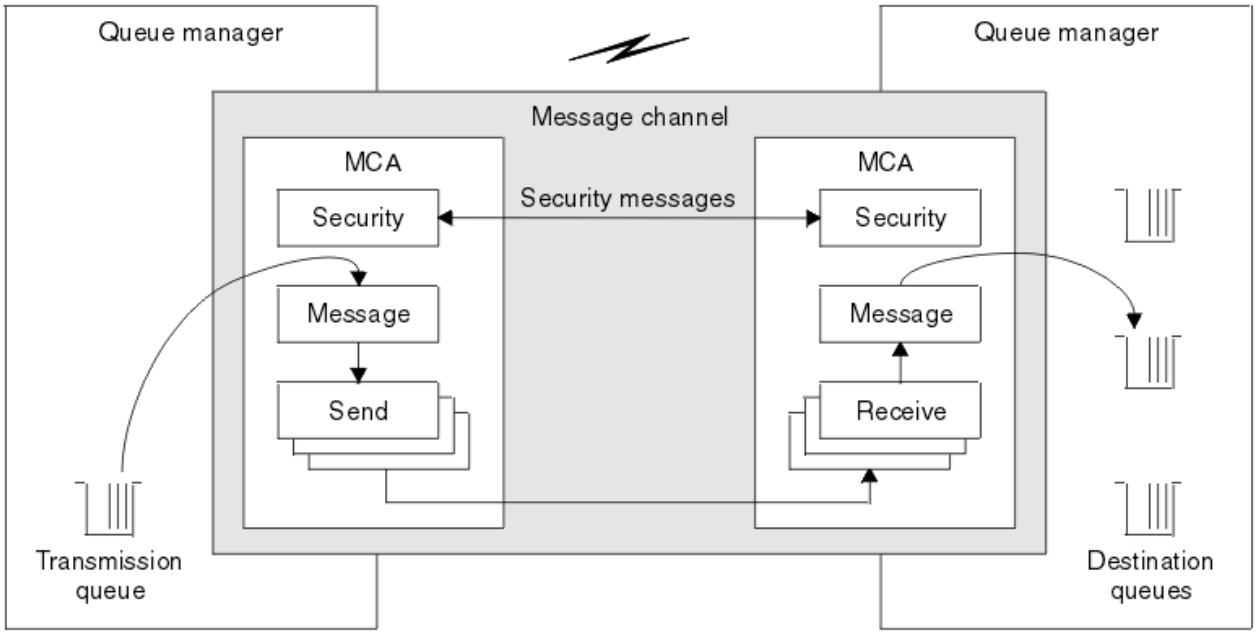


Figure 11. Exits de sécurité, de message, d'envoi et de réception sur un canal de message

Concepts associés

Programmes d'exit de canal pour les canaux de messagerie

Présentation de l'exit de sécurité

Les exits de sécurité fonctionnent normalement par paires. Ils sont appelés avant le flux de messages et leur but est de permettre à un agent MCA d'authentifier son partenaire.

Les *exits de sécurité* fonctionnent normalement par paires, une à chaque extrémité d'un canal. Ils sont appelés immédiatement après la fin de la négociation de données initiale au démarrage du canal, mais avant que les messages ne commencent à circuler. L'objectif principal de l'exit de sécurité est d'activer l'agent MCA à chaque extrémité d'un canal pour authentifier son partenaire. Cependant, rien n'empêche un exit de sécurité d'exécuter une autre fonction, même une fonction qui n'a rien à voir avec la sécurité.

Les exits de sécurité peuvent communiquer entre eux en envoyant des *messages de sécurité*. Le format du message de sécurité est défini par l'utilisateur. Un résultat possible de l'échange de messages de sécurité est que l'un des exits de sécurité peut décider de ne pas poursuivre. Dans ce cas, le canal est fermé et les messages ne circulent pas. S'il n'y a un exit de sécurité qu'à une seule extrémité d'un canal, l'exit est toujours appelé et peut choisir de continuer ou de fermer le canal.

Les exits de sécurité peuvent être appelés sur les canaux de message et MQI. Le nom d'un exit de sécurité est spécifié en tant que paramètre dans la définition de canal à chaque extrémité d'un canal.

Pour plus d'informations sur les exits de sécurité, voir «Sécurité au niveau de la liaison à l'aide d'un exit de sécurité», à la page 105.

Exit de message

Les exits de message fonctionnent uniquement sur les canaux de message et fonctionnent normalement par paires. Un exit de message peut fonctionner sur l'ensemble du message et y apporter diverses modifications.

Les exits de message aux extrémités émettrice et réceptrice d'un canal fonctionnent normalement par paires. Un exit de message à l'extrémité émettrice d'un canal est appelé une fois que l'agent MCA a reçu un message de la file d'attente de transmission. A l'extrémité réceptrice d'un canal, un exit de message est appelé avant que l'agent MCA n'insère un message dans sa file d'attente de destination.

Un exit de message a accès à l'en-tête de file d'attente de transmission, MQXQH, qui inclut le descripteur de message imbriqué, et aux données d'application d'un message. Un exit de message peut modifier le contenu du message et modifier sa longueur. Un changement de longueur peut être le résultat de la

compression, de la décompression, du chiffrement ou du déchiffrement du message. Il peut également être le résultat de l'ajout de données au message ou de la suppression de données de celui-ci.

Les exits de message peuvent être utilisés à n'importe quelle fin qui nécessite l'accès à l'ensemble du message, plutôt qu'à une partie de celui-ci, et pas nécessairement pour des raisons de sécurité.

Un exit de message peut déterminer que le message qu'il est en train de traiter ne doit pas continuer vers sa destination. L'agent MCA place ensuite le message dans la file d'attente des messages non livrés. Un exit de message peut également fermer le canal.

Les exits de message peuvent être appelés uniquement sur les canaux de message et non sur les canaux MQI. En effet, l'objectif d'un canal MQI est d'activer les paramètres d'entrée et de sortie des appels MQI entre l'application IBM MQ MQI client et le gestionnaire de files d'attente.

Le nom d'un exit de message est indiqué en tant que paramètre dans la définition de canal à chaque extrémité d'un canal. Vous pouvez également spécifier une liste d'exits de message à exécuter successivement.

Pour plus d'informations sur les exits de message, voir «Sécurité au niveau de la liaison à l'aide d'un exit de message», à la page 105.

Exits d'envoi et de réception

Les exits d'envoi et de réception fonctionnent généralement par paires. Ils fonctionnent sur des segments de transmission et sont utilisés au mieux lorsque la structure des données qu'ils traitent n'est pas pertinente.

Un *exit d'émission* à une extrémité d'un canal et un *exit de réception* à l'autre extrémité fonctionnent normalement par paires. Un exit d'émission est appelé juste avant qu'un agent MCA ne lance un envoi de communications pour envoyer des données via une connexion de communication. Un exit de réception est appelé juste après qu'un agent MCA a repris le contrôle à la suite d'une réception de communications et a reçu des données d'une connexion de communication. Si le partage de conversations est en cours d'utilisation, sur un canal MQI, une instance différente d'exit d'émission et de réception est appelée pour chaque conversation.

Les flux du protocole de canal IBM MQ entre deux agents MCA sur un canal de transmission de messages contiennent des informations de contrôle ainsi que des données de message. De même, sur un canal MQI, les flux contiennent des informations de contrôle ainsi que les paramètres des appels MQI. Les exits d'envoi et de réception sont appelés pour tous les types de données.

Les données de message ne circulent que dans une seule direction sur un canal de message mais, sur un canal MQI, les paramètres d'entrée d'un flux d'appel MQI dans une direction et les paramètres de sortie dans l'autre. Sur les canaux de message et MQI, contrôlez les flux d'informations dans les deux sens. Par conséquent, les exits d'émission et de réception peuvent être appelés aux deux extrémités d'un canal.

L'unité de données qui est transmise dans un flux unique entre deux MCM est appelée *segment de transmission*. Les exits d'émission et de réception ont accès à chaque segment de transmission. Ils peuvent modifier son contenu et sa longueur. Toutefois, un exit d'émission ne doit pas modifier les 8 premiers octets d'un segment de transmission. Ces 8 octets font partie de l'en-tête de protocole de canal IBM MQ . Il existe également des restrictions sur la mesure dans laquelle un exit d'émission peut augmenter la longueur d'un segment de transmission. En particulier, un exit d'émission ne peut pas augmenter sa longueur au-delà de la longueur maximale négociée entre les deux agents MCA au démarrage du canal.

Sur un canal de transmission, si un message est trop volumineux pour être envoyé dans un seul segment de transmission, l'agent MCA émetteur fractionne le message et l'envoie dans plusieurs segments de transmission. En conséquence, une sortie d'émission est appelée pour chaque segment de transmission contenant une partie du message et, à la réception, une sortie de réception est appelée pour chaque segment de transmission. L'agent MCA récepteur reconstitue le message des segments de transmission après qu'ils ont été traités par l'exit de réception.

De même, sur un canal MQI, les paramètres d'entrée ou de sortie d'un appel MQI sont envoyés dans plusieurs segments de transmission s'ils sont trop grands. Cela peut se produire, par exemple, sur un appel MQPUT, MQPUT1ou MQGET si les données d'application sont suffisamment volumineuses.

Compte tenu de ces considérations, il est plus approprié d'utiliser des exits d'émission et de réception à des fins dans lesquelles ils n'ont pas besoin de comprendre la structure des données qu'ils traitent et peuvent donc traiter chaque segment de transmission comme un objet binaire.

Un exit d'émission ou de réception peut fermer un canal.

Les noms d'un exit d'émission et d'un exit de réception sont spécifiés en tant que paramètres dans la définition de canal à chaque extrémité d'un canal. Vous pouvez également spécifier une liste d'exits d'émission à exécuter successivement. De même, vous pouvez spécifier une liste d'exits de réception.

Pour plus d'informations sur les exits d'envoi et de réception, voir «Sécurité au niveau de la liaison à l'aide des exits d'envoi et de réception», à la page 105.

Planification de l'intégrité des données

Planifiez la manière de préserver l'intégrité de vos données.

Vous pouvez implémenter l'intégrité des données au niveau de l'application ou du lien.

Au niveau de l'application, vous pouvez utiliser des programmes d'exit API si les fonctions standard ne répondent pas à vos besoins. Vous pouvez choisir d'utiliser Advanced Message Security (AMS) pour signer numériquement des messages afin de vous protéger contre les modifications non autorisées.

Au niveau des liens, vous pouvez choisir d'utiliser TLS, auquel cas vous devez planifier votre utilisation des certificats numériques. Vous pouvez également utiliser des programmes d'exit de canal si les fonctions standard ne répondent pas à vos besoins.

Concepts associés

«Protection des canaux avec SSL/TLS», à la page 114
La prise en charge de TLS dans IBM MQ utilise l'objet d'informations d'authentification du gestionnaire de files d'attente et diverses commandes MQSC. Vous devez également tenir compte de votre utilisation des certificats numériques.

«Intégrité des données dans IBM MQ», à la page 23
Vous pouvez utiliser un service d'intégrité des données pour détecter si un message a été modifié.

«Planification de Advanced Message Security», à la page 106
Advanced Message Security (AMS) est un composant de IBM MQ qui offre un niveau de protection élevé pour les données sensibles transitant par le réseau IBM MQ , sans affecter les applications finales.

Structures de données et appels d'exit de canal

Référence associée

Référence d'exit API

Planification de l'audit

Décidez des données à auditer et de la manière dont vous allez capturer et traiter les informations d'audit. Vérifiez que votre système est correctement configuré.

La surveillance de l'activité comporte plusieurs aspects. Les aspects que vous devez prendre en compte sont souvent définis par des exigences d'auditeur, et ces exigences sont souvent dictées par des normes réglementaires telles que la loi HIPAA (Health Insurance Portability and Accountability Act) ou la loi SOX (Sarbanes-Oxley). IBM MQ fournit des fonctions destinées à faciliter la conformité à ces normes.

Déterminez si vous êtes intéressé uniquement par les exceptions ou si vous êtes intéressé par tous les comportements du système.

Certains aspects de l'audit peuvent également être considérés comme une surveillance opérationnelle ; une distinction pour l'audit est que vous examinez souvent les données historiques, et pas seulement les alertes en temps réel. La surveillance est traitée dans la section Surveillance et performances.

Données à auditer

Prenez en compte les types de données ou d'activité que vous devez auditer, comme décrit dans les sections suivantes:

Modifications apportées à IBM MQ à l'aide des interfaces IBM MQ
Configurez IBM MQ pour émettre des événements d'instrumentation, en particulier des événements de commande et des événements de configuration.

Modifications apportées à IBM MQ en dehors de son contrôle
Certaines modifications peuvent affecter le comportement de IBM MQ , mais elles ne peuvent pas être directement surveillées par IBM MQ. Par exemple, vous pouvez modifier les fichiers de configuration `mqs.ini`, `qm.ini`et `mqclient.ini`, créer et supprimer des gestionnaires de files d'attente, installer des fichiers binaires tels que des programmes d'exit utilisateur et modifier les droits d'accès aux fichiers. Pour surveiller ces activités, vous devez utiliser des outils exécutés au niveau du système d'exploitation. Différents outils sont disponibles et adaptés aux différents systèmes d'exploitation. Vous pouvez également avoir des journaux créés par des outils associés, tels que *sudo*.

Contrôle opérationnel de IBM MQ
Vous devrez peut-être utiliser les outils du système d'exploitation pour auditer les activités telles que le démarrage et l'arrêt des gestionnaires de files d'attente. Dans certains cas, IBM MQ peut être configuré pour émettre des événements d'instrumentation.

Activité d'application dans IBM MQ
Pour auditer les actions des applications, par exemple l'ouverture de files d'attente et l'insertion et l'obtention de messages, configurez IBM MQ pour émettre des événements appropriés.

Alertes d'intrus
Pour auditer les tentatives d'atteinte à la sécurité, configurez votre système pour qu'il émet des événements d'autorisation. Les événements de canal peuvent également être utiles pour afficher l'activité, en particulier si un canal se termine de manière inattendue.

Planification de la capture, de l'affichage et de l'archivage des données d'audit

La plupart des éléments dont vous avez besoin sont signalés comme des messages d'événement IBM MQ . Vous devez choisir des outils qui peuvent lire et mettre en forme ces messages. Si vous êtes intéressé par le stockage et l'analyse à long terme, vous devez les déplacer vers un mécanisme de mémoire secondaire tel qu'une base de données. Si vous ne traitez pas ces messages, ils restent dans la file d'attente d'événements, ce qui peut entraîner le remplissage de la file d'attente. Vous pouvez décider d'implémenter un outil qui exécute automatiquement des actions en fonction de certains événements ; par exemple, pour émettre une alerte lorsqu'un incident de sécurité se produit.

Vérification de la configuration correcte de votre système

Un ensemble de tests est fourni avec IBM MQ Explorer. Utilisez ces éléments pour rechercher les problèmes éventuels dans les définitions d'objet.

Vérifiez également régulièrement que la configuration du système correspond à vos attentes. Bien que les événements de commande et de configuration puissent signaler une modification, il est également utile de vider la configuration et de la comparer à une copie correcte connue.

Planification de la sécurité par topologie

Cette section traite de la sécurité dans des situations spécifiques, à savoir pour les canaux, les clusters de gestionnaires de files d'attente, les applications de publication / abonnement et de multidiffusion, et lors de l'utilisation d'un pare-feu.

Pour plus d'informations, voir les sous-rubriques suivantes:

Autorisation de canal

Lorsque vous envoyez ou recevez un message via un canal, vous devez fournir un accès à diverses ressources IBM MQ . Les agents MCA (Message Channel Agent) sont essentiellement des applications IBM MQ qui déplacent les messages entre les gestionnaires de files d'attente et qui, en tant que telles, nécessitent un accès à diverses ressources IBM MQ pour fonctionner correctement.

Pour recevoir des messages au moment de l'opération PUT pour les agents MCA, vous pouvez utiliser l'ID utilisateur associé à l'agent MCA ou l'ID utilisateur associé au message.

Au moment de la connexion, vous pouvez mapper l'ID utilisateur vérifié à un autre utilisateur, à l'aide des enregistrements d'authentification de canal **CHLAUTH** .

Dans IBM MQ, les canaux peuvent être protégés par le support TLS.

Les ID utilisateur associés aux canaux d'envoi et de réception, à l'exception du canal émetteur où l'attribut MCAUSER n'est pas utilisé, requièrent l'accès aux ressources suivantes:

- L'ID utilisateur associé à un canal émetteur requiert l'accès au gestionnaire de files d'attente, à la file d'attente de transmission, à la file d'attente de rebut et à toute autre ressource requise par les exits de canal.
- L'ID utilisateur MCAUSER d'un canal récepteur requiert les droits *+ setall* . En effet, le canal récepteur doit créer le MQMD complet, y compris toutes les zones de contexte, à l'aide des données qu'il a reçues du canal émetteur distant. Le gestionnaire de files d'attente requiert donc que l'utilisateur exécutant cette activité dispose des droits *+ setall* . Ces droits *+ setall* doivent être accordés à l'utilisateur pour:
  - Toutes les files d'attente dans lesquelles le canal récepteur insère des messages de manière valide.
  - Objet gestionnaire de files d'attente. Pour plus d'informations, voir Autorisations de contexte.
- L'ID utilisateur MCAUSER d'un canal récepteur sur lequel l'émetteur a demandé un message de rapport COA requiert le droit *+ passid* sur la file d'attente de transmission qui renvoie le message de rapport. Sans ces droits, les messages d'erreur AMQ8077 sont consignés.
- Avec l'ID utilisateur associé au canal récepteur, vous pouvez ouvrir les files d'attente cible pour y placer des messages. Cela implique l'interface MQI (Message queuing Interface), de sorte que des vérifications de contrôle d'accès supplémentaires peuvent être nécessaires si vous n'utilisez pas IBM MQ Object Authority Manager (OAM). Vous pouvez indiquer si les vérifications d'autorisation sont effectuées sur l'ID utilisateur associé à l'agent MCA (comme décrit dans cette rubrique) ou sur l'ID utilisateur associé au message (à partir de la zone MQMD UserIdentifier).

  Pour les types de canal auxquels il s'applique, le paramètre **PUTAUT** d'une définition de canal indique l'ID utilisateur utilisé pour ces vérifications.
  - Par défaut, le canal utilise le compte de service du gestionnaire de files d'attente, qui dispose de droits d'administration complets et ne requiert aucune autorisation spéciale.
  - Dans le cas des canaux de connexion serveur, les connexions d'administration sont bloquées par défaut par les règles CHLAUTH et nécessitent une mise à disposition explicite.
  - Les canaux de type récepteur, demandeur et récepteur de cluster permettent l'administration locale par tout gestionnaire de files d'attente adjacent, sauf si l'administrateur prend des mesures pour restreindre cet accès.
  - Il n'est pas nécessaire d'accorder les droits *dsp* et *ctrlx* pour l'ID utilisateur MCAUSER d'un canal récepteur.
  - Avant IBM MQ 8.0.0 Fix Pack 4, si vous utilisez un ID utilisateur qui ne dispose pas de privilèges d'administration IBM MQ , vous devez accorder les droits **dsp** et **ctrlx** pour le canal à cet ID utilisateur pour que le canal fonctionne.

    Depuis la IBM MQ 8.0.0 Fix Pack 4, il n'y a pas de contrôle des droits d'accès lorsqu'un canal se resynchronise et corrige les numéros de séquence.

    Toutefois, l'émission manuelle d'une commande RESET CHANNEL requiert toujours **+dsp** et **+ctrlx** dans toutes les éditions.

    Avertissement : Lorsqu'une réinitialisation de canal est nécessaire pour la confirmation par lots de messages, IBM MQ tente d'interroger le canal, ce qui nécessite des droits d'accès **+dsp** .
  - L'attribut MCAUSER n'est pas utilisé pour le type de canal SDR.
  - Si vous utilisez l'ID utilisateur associé au message, il est probable que l'ID utilisateur provient d'un système distant. Cet ID utilisateur de système distant doit être reconnu par le système cible. Les

commandes suivantes sont des exemples du type de commande que vous pouvez exécuter pour accorder des droits à un ID utilisateur à partir d'un système distant:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect +inq +setall
```

```
setmqaut -m QMgrName -t chl -n Profile -g GroupName +dsp +ctrlx
```

où *Profil* est un canal.

```
setmqaut -m QMgrName -t q -n Profile -g GroupName +put +setall
```

où *Profil* est une file d'attente de rebut, si elle est définie.

```
setmqaut -m QMgrName -t q -n Profile -g GroupName +put +setall
```

où *Profil* est une liste de files d'attente autorisées.

Avertissement : Soyez prudent lorsque vous autorisez un ID utilisateur à placer des messages dans la file d'attente de commandes ou dans d'autres files d'attente système sensibles.

L'ID utilisateur associé à l'agent MCA dépend du type d'agent MCA. Il existe deux types d'agent MCA:

Agent MCA appelant
Les agents MCA qui initient un canal. Les agents MCA appelants peuvent être démarrés en tant que processus individuels, en tant qu'unités d'exécution de l'initiateur de canal ou en tant qu'unités d'exécution d'un pool de processus. L'ID utilisateur utilisé est l'ID utilisateur associé au processus parent (initiateur de canal) ou l'ID utilisateur associé au processus qui démarre l'agent MCA.

Agent MCA répondeur
Les agents MCA répondeurs sont des agents MCA démarrés à la suite d'une demande d'un agent MCA appelant. Les agents MCA répondeurs peuvent être démarrés en tant que processus individuels, en tant qu'unités d'exécution du programme d'écoute ou en tant qu'unités d'exécution d'un pool de processus. L'ID utilisateur peut être l'un des types suivants (dans cet ordre de préférence):

1. Sur APPC, l'agent MCA appelant peut indiquer l'ID utilisateur à utiliser pour l'agent MCA répondeur. Cet ID est appelé ID utilisateur réseau et s'applique uniquement aux canaux démarrés en tant que processus individuels. Définissez l'ID utilisateur réseau à l'aide du paramètre USERID de la définition de canal.
2. Si le paramètre **USERID** n'est pas utilisé, la définition de canal de l'agent MCA répondeur peut indiquer l'ID utilisateur que l'agent MCA doit utiliser. Définissez l'ID utilisateur à l'aide du paramètre **MCAUSER** de la définition de canal.
3. Si l'ID utilisateur n'a été défini par aucune des deux méthodes précédentes, l'ID utilisateur du processus qui démarre l'agent MCA ou l'ID utilisateur du processus parent (le programme d'écoute) est utilisé.

Concepts associés

«Enregistrements d'authentification de canal», à la page 50
Pour exercer un contrôle plus précis sur les accès accordés aux systèmes en cours de connexion au niveau d'un canal, vous pouvez utiliser les enregistrements d'authentification de canal.

Propriétés de l'enregistrement d'authentification de canal

Protection des définitions d'initialisateur de canal

Seuls les membres du groupe mqm peuvent manipuler les initiateurs de canal.

Les initiateurs de canal IBM MQ ne sont pas des objets IBM MQ ; leur accès n'est pas contrôlé par la méthode d'accès aux objets (OAM). IBM MQ n'autorise pas les utilisateurs ou les applications à manipuler ces objets, sauf si leur ID utilisateur est membre du groupe mqm. Si vous disposez d'une application qui émet la commande PCF **StartChannelInitiator**, l'ID utilisateur spécifié dans le descripteur de message du message PCF doit être membre du groupe mqm sur le gestionnaire de files d'attente cible.

Un ID utilisateur doit également être membre du groupe mqm sur la machine cible pour émettre les commandes MQSC équivalentes via la commande Escape PCF ou à l'aide de `runmqsc` en mode indirect.

Files d'attente de transmission

Les gestionnaires de files d'attente placent automatiquement les messages éloignés dans une file d'attente de transmission ; aucun droit spécial n'est requis à cet effet.

Toutefois, si vous devez placer un message directement dans une file d'attente de transmission, vous devez disposer d'une autorisation spéciale ; voir Tableau 12, à la page 132.

Exits de canal

Si les enregistrements d'authentification de canal ne conviennent pas, vous pouvez utiliser des exits de canal pour une sécurité accrue. Un exit de sécurité établit une connexion sécurisée entre deux programmes d'exit de sécurité. Un programme est destiné à l'agent MCA émetteur et un programme est destiné à l'agent MCA récepteur.

Pour plus d'informations sur les exits de canal, voir «Programmes d'exit de canal», à la page 107 .

Protection des canaux avec SSL/TLS

La prise en charge de TLS dans IBM MQ utilise l'objet d'informations d'authentification du gestionnaire de files d'attente et diverses commandes MQSC. Vous devez également tenir compte de votre utilisation des certificats numériques.

Certificats numériques et référentiels de clés

Il est recommandé de définir l'attribut de label de certificat du gestionnaire de files d'attente (**CERTLABL**) au nom du certificat personnel à utiliser pour la majorité des canaux, et le remplacer pour les exceptions, en définissant le label de certificat sur les canaux qui requièrent des certificats différents.

Si vous avez besoin de plusieurs canaux avec des certificats qui diffèrent du certificat par défaut défini sur le gestionnaire de files d'attente, vous devez envisager de diviser les canaux entre plusieurs gestionnaires de files d'attente ou d'utiliser un proxy MQIPT devant le gestionnaire de files d'attente pour présenter un certificat différent.

Vous pouvez utiliser un certificat différent pour chaque canal, mais si vous stockez un trop grand nombre de certificats dans un référentiel de clés, vous pouvez vous attendre à ce que les performances soient affectées lors du démarrage des canaux TLS. Essayez de maintenir le nombre de certificats dans un référentiel de clés à moins de 50 et considérez que 100 est un maximum car les performances de GSKit diminuent fortement avec les référentiels de clés plus volumineux.

L'autorisation de plusieurs certificats sur le même gestionnaire de files d'attente augmente les chances que plusieurs certificats de l'autorité de certification soient utilisés sur le même gestionnaire de files d'attente. Cela augmente les chances de conflits d'espace de nom de nom distinctif de sujet de certificat pour les certificats émis par des autorités de certification distinctes.

Alors que les autorités de certification professionnelles sont susceptibles d'être plus prudentes, les autorités de certification internes manquent souvent de conventions de dénomination claires et vous pourriez vous retrouver avec des correspondances inattendues entre une autorité de certification et une autre.

Vous devez vérifier le nom distinctif de l'émetteur du certificat en plus du nom distinctif du sujet. Pour ce faire, utilisez un enregistrement SSLPEERMAP d'authentification de canal et définissez les zones **SSLPEER** et **SSLCERTI** pour qu'elles correspondent respectivement au nom distinctif du sujet et au nom distinctif de l'émetteur.

Certificats autosignés et signés par une autorité de certification

Il est important de planifier votre utilisation des certificats numériques, à la fois lorsque vous développez et testez votre application, et pour son utilisation en production. Vous pouvez utiliser des certificats signés par une autorité de certification ou des certificats autosignés, en fonction de l'utilisation des gestionnaires de files d'attente et des applications client.

Certificats signés par une autorité de certification
Pour les systèmes de production, procurez-vous vos certificats auprès d'une autorité de certification digne de confiance. Lorsque vous obtenez un certificat d'une autorité de certification externe, vous payez pour le service.

certificats autosignés
Lors du développement de votre application, vous pouvez utiliser des certificats autosignés ou des certificats émis par une autorité de certification locale, en fonction de la plateforme:

ULW Sur les systèmes Windows, UNIXet Linux , vous pouvez utiliser des certificats autosignés. Voir «Création d'un certificat personnel autosigné sur UNIX, Linux, and Windows», à la page 303 pour des instructions.

IBM i Sur les systèmes IBM i , vous pouvez utiliser des certificats signés par l'autorité de certification locale. Voir «Demande d'un certificat serveur sous IBM i», à la page 286 pour des instructions.

z/OS Sous z/OS, vous pouvez utiliser des certificats autosignés ou des certificats signés par une autorité de certification locale. Pour obtenir des instructions, voir «Création d'un certificat personnel autosigné sur z/OS», à la page 331 ou «Demande d'un certificat personnel sur z/OS», à la page 331 .

Les certificats autosignés ne conviennent pas à une utilisation en production pour les raisons suivantes :

- Les certificats autosignés ne peuvent pas être révoqués, ce qui peut permettre à un agresseur de usurper une identité après qu'une clé privée a été compromise. Les autorités de certification peuvent révoquer un certificat compromis, pour en empêcher toute utilisation future. Les certificats signés par une autorité de certification sont donc plus sûrs à utiliser dans un environnement de production, bien que les certificats autosignés soient plus pratiques pour un système de test.
- Les certificats autosignés n'arrivent jamais à expiration. Ce comportement est pratique et sûr dans un environnement de test, mais dans un environnement de production, les certificats restent ouverts et donc sujets à des violations de sécurité. Ce risque est aggravé du fait que les certificats autosignés ne peuvent pas être révoqués.
- Un certificat autosigné est utilisé à la fois comme certificat personnel et comme certificat d'autorité de certification racine (ou ancrage sécurisé). Un utilisateur avec un certificat personnel autosigné doit pouvoir l'utiliser pour signer d'autres certificats personnels. En général, cela n'est pas vrai des certificats personnels émis par une autorité de certification et représente un risque important.

CipherSpecs et certificats numériques

Seul un sous-ensemble des CipherSpecs pris en charge peut être utilisé avec tous les types de certificat numérique pris en charge. Il est donc nécessaire de choisir un CipherSpec approprié pour vos certificats numériques. De même, si la stratégie de sécurité de votre organisation requiert l'utilisation d'un CipherSpec particulier, vous devez obtenir des certificats numériques appropriés.

Pour plus d'informations sur la relation entre les CipherSpecs et les certificats numériques, voir «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45

Règles de validation de certificat

La norme IETF RFC 5280 spécifie une série de règles de validation de certificat que les logiciels d'application conformes doivent implémenter afin d'éviter les attaques d'usurpation d'identité. Un ensemble de règles de validation de certificat est appelé règle de validation de certificat. Pour plus d'informations sur les règles de validation de certificat dans IBM MQ, voir «Règles de validation de certificat dans IBM MQ», à la page 44.

Planification de la vérification de la révocation de certificat

L'autorisation de plusieurs certificats provenant de différentes autorités de certification peut entraîner une vérification supplémentaire inutile de la révocation des certificats.

En particulier, si vous avez configuré explicitement l'utilisation d'un serveur de révocation d'une autorité de certification particulière, par exemple en utilisant un objet AUTHINFO ou une structure d'enregistrement d'informations d'authentification (MQAIR), une vérification de révocation échoue lorsqu'elle est présentée avec un certificat d'une autre autorité de certification.

Vous devez éviter la configuration explicite du serveur de révocation de certificat. Au lieu de cela, vous devez activer la vérification implicite lorsque chaque certificat contient son propre emplacement de serveur de révocation dans une extension de certificat, par exemple, CRL Distribution Point ou OCSP AuthorityInfoAccess.

Pour plus d'informations, voir OCSPCheckExtensions et CDPCheckExtensions.

Commandes et attributs pour la prise en charge de TLS

Le protocole TLS (Transport Layer Security) fournit une sécurité de canal, avec une protection contre les écoutes clandestines, les falsifications et les usurpations d'identité. La prise en charge de TLS par IBM MQ vous permet de spécifier, dans la définition de canal, qu'un canal particulier utilise la sécurité TLS. Vous pouvez également spécifier les détails du type de sécurité de votre choix, par exemple l'algorithme de chiffrement que vous souhaitez utiliser.

- Les commandes MQSC suivantes prennent en charge TLS:

  ALTER AUTHINFO
  Modifie les attributs d'un objet d'informations d'authentification.

  DEFINE AUTHINFO
  Crée un objet d'informations d'authentification.

  DELETE AUTHINFO
  Supprime un objet d'informations d'authentification.

  INFORMATIONS D'AUTHENTIFICATION D'AFFICHAGE
  Affiche les attributs d'un objet d'informations d'authentification spécifique.

- Les paramètres de gestionnaire de files d'attente suivants prennent en charge TLS:

  CERTLABL
  Définit un libellé de certificat personnel à utiliser.

  SSLCRLNL
  L'attribut SSLCRLNL spécifie une liste de noms d'objets d'informations d'authentification qui sont utilisés pour fournir des emplacements de révocation de certificat afin de permettre une vérification améliorée des certificats TLS.

  SSLCRYP
  Sur les systèmes Windows , UNIX and Linux , définit l'attribut de gestionnaire de files d'attente **SSLCryptoHardware** . Cet attribut est le nom de la chaîne de paramètres que vous pouvez utiliser pour configurer le matériel cryptographique que vous avez sur votre système.

  SSLEV
  Détermine si un message d'événement TLS est signalé si un canal utilisant TLS ne parvient pas à établir une connexion TLS.

  SSLFIPS
  Indique si seuls les algorithmes certifiés FIPS doivent être utilisés si la cryptographie est effectuée dans IBM MQ , plutôt que dans le matériel de cryptographie. Si le matériel de cryptographie est configuré, les modules de cryptographie fournis par le produit matériel sont utilisés et ceux-ci peuvent être certifiés FIPS à un niveau particulier. Cela dépend du produit matériel utilisé.

  SSLKEYR
  Sur les systèmes UNIX, Linux, and Windows , associe un référentiel de clés à un gestionnaire de files d'attente. La base de données de clés est conservée dans une base de données de clés *GSKit* . IBM

Global Security Kit (GSKit) vous permet d'utiliser la sécurité TLS sur les systèmes Windows , UNIX and Linux .

SSLRKEYC
Nombre d'octets à envoyer et à recevoir dans une conversation TLS avant la renégociation de la clé secrète. Le nombre d'octets inclut les informations de contrôle envoyées par l'agent MCA.

- Les paramètres de canal suivants prennent en charge TLS:

CERTLABL
Définit un libellé de certificat personnel à utiliser.

SSLCAUTH
Indique si IBM MQ requiert et valide un certificat du client TLS.

SSLCIPH
Indique la force et la fonction de chiffrement (CipherSpec), par exemple TLS_RSA_WITH_AES_128_CBC_SHA. Le CipherSpec doit correspondre aux deux extrémités du canal.

SSLPEER
Indique le nom distinctif (identificateur unique) des partenaires autorisés.

Cette section décrit les commandes **setmqaut**, **dspmqaut**, **dmpmqaut**, **rcrmqobj**, **rcdmqimg**et **dspmqfls** permettant de prendre en charge l'objet d'informations d'authentification. Il décrit également la commande **runmqckm** (iKeycmd) pour la gestion des certificats sur les systèmes UNIX and Linux et l'outil **runmqakm** pour la gestion des certificats sur UNIX, Linux, and Windows. Reportez-vous aux sections suivantes :

- setmqaut
- dspmqaut
- dmpmqaut
- rcrmqobj
- rcdmqimg
- dspmqfls
- Gestion des clés et des certificats

Pour une présentation de la sécurité des canaux à l'aide de TLS, voir

- «Protocoles de sécurité TLS dans IBM MQ», à la page 24

Pour plus de détails sur les commandes MQSC associées à TLS, voir

- ALTER AUTHINFO
- DEFINE AUTHINFO
- DELETE AUTHINFO
- DISPLAY AUTHINFO

Pour plus de détails sur les commandes PCF associées à TLS, voir

- Change, Copy, and Create Authentication Information Object
- Delete Authentication Information Object
- Inquire Authentication Information Object

z/OS *IBM MQ for z/OS Canal de connexion serveur*

Le canal IBM MQ for z/OS SVRCONN n'est pas sécurisé sans l'implémentation de l'authentification de canal ou l'ajout d'un exit de sécurité à l'aide de TLS. Les canaux SVRCONN n'ont pas d'exit de sécurité défini par défaut.

Problèmes de sécurité

Les canaux SVRCONN ne sont pas sécurisés comme initialement définis, SYSTEM.DEF.SVRCONN par exemple. Pour sécuriser un canal SVRCONN, vous devez configurer l'authentification de canal à l'aide de la commande SET CHLAUTH ou installer un exit de sécurité et implémenter TLS.

Vous devez utiliser un exemple d'exit de sécurité accessible au public, écrire vous-même un exit de sécurité ou acheter un exit de sécurité.

Il existe plusieurs exemples que vous pouvez utiliser comme point de départ pour l'écriture de votre propre exit de sécurité de canal SVRCONN.

Dans IBM MQ for z/OS, le membre CSQ4BCX3 de votre bibliothèque hlq.SCSQC37S est un exemple d'exit de sécurité écrit en langage C. L'exemple CSQ4BCX3 est également fourni dans votre bibliothèque hlq.SCSQAUTH .

Vous pouvez implémenter l'exemple d'exit CSQ4BCX3 en copiant le membre compilé hlq.SCSQAUTH(CSQ4BCX3) dans une bibliothèque de chargement allouée à la définition de données CSQXLIB dans votre procédure CHIN. Notez que le code CHIN requiert que la bibliothèque de chargement soit définie comme "Program Controlled".

Modifiez votre canal SVRCONN pour définir CSQ4BCX3 comme exit de sécurité.

V 9.1.4 Lorsqu'un client se connecte à l'aide de ce canal SVRCONN, CSQ4BCX3 s'authentifie à l'aide de la paire **RemoteUserIdentifier** et **RemotePassword** à partir de MQCD ou, à partir de IBM MQ 9.1.4, de la paire **CSPUserIdPtr** et **CSPPasswordPtr** à partir de MQCSP. Si l'authentification aboutit, elle copie **RemoteUserIdentifier** dans **MCAUserIdentifier**, en modifiant le contexte d'identité de l'unité d'exécution.

Pour Long Term Support et Continuous Delivery avant IBM MQ 9.1.4, lorsqu'un client se connecte à l'aide de ce canal SVRCONN, CSQ4BCX3 s'authentifie à l'aide de la paire **RemoteUserIdentifier** et **RemotePassword** de MQCD. Si l'authentification aboutit, elle copie **RemoteUserIdentifier** dans **MCAUserIdentifier**, en modifiant le contexte d'identité de l'unité d'exécution.

Si vous écrivez un client IBM MQ Java , vous pouvez utiliser des fenêtres en incrustation pour interroger l'utilisateur et définir MQEnvironment.userID et MQEnvironment.password. Ces valeurs seront transmises lorsque la connexion sera établie.

Maintenant que vous disposez d'un exit de sécurité fonctionnel, il est à craindre que l'ID utilisateur et le mot de passe soient transmis en texte en clair sur le réseau lorsque la connexion est établie, tout comme le contenu des messages IBM MQ suivants. Vous pouvez utiliser TLS pour chiffrer ces informations de connexion initiale ainsi que le contenu des messages IBM MQ .

Exemple

Pour sécuriser le IBM MQ Explorer canal SVRCONN SYSTEM.ADMIN.SVRCONN :

1. Copiez hlq.SCSQAUTH(CSQ4BCX3) dans une bibliothèque de chargement allouée à la définition de données CSQXLIB dans la procédure CHINIT.
2. Vérifiez que la bibliothèque de chargement est contrôlée par programme.
3. Modifiez SYSTEM ADMIN.SVRCONN pour utiliser l'exit de sécurité CSQ4BCX3.
4. Dans IBM MQ Explorer, cliquez avec le bouton droit de la souris sur le nom du gestionnaire de files d'attente z/OS , sélectionnez **Détails de la connexion** > **Propriétés** > **ID utilisateur** et entrez votre ID utilisateur z/OS .
5. Connectez-vous au gestionnaire de files d'attente z/OS en entrant un mot de passe.

Renseignements supplémentaires

Pour que l'exit CSQ4BCX3 s'exécute dans un environnement contrôlé par programme, tous les éléments chargés dans l'espace adresse CHIN doivent être chargés à partir d'une bibliothèque contrôlée par programme, par exemple, toutes les bibliothèques dans STEPLIB et toutes les bibliothèques nommées

dans CSQXLIB DD. Pour définir une bibliothèque de chargement comme étant contrôlée par le programme, exécutez les commandes RACF . Dans l'exemple suivant, le nom de la bibliothèque de chargement est MY.TEST.LOADLIB.

```
RALTER PROGRAM * ADDMEM('MY.TEST.LOADLIB'//NOPADCHK)
SETROPTS WHEN(PROGRAM)REFRESH
```

Pour modifier le canal SVRCONN afin d'implémenter CSQ4BCX3, exécutez la commande IBM MQ suivante:

```
ALTER CHANNEL(SYSTEM ADMIN.SVRCONN) CHLTYPE(SVRCONN) SCYEXIT(CSQ4BCX3)
```

Dans l'exemple ci-dessus, le nom de canal SVRCONN utilisé est SYSTEM ADMIN.SVRCONN.

Pour plus d'informations sur les exits de canal, voir «Programmes d'exit de canal», à la page 107 .

Tâches associées
Ecriture de programmes d'exit de canal sous z/OS

Services de sécurité SNA LU 6.2

L'unité logique SNA 6.2 offre la cryptographie au niveau de la session, l'authentification au niveau de la session et l'authentification au niveau de la conversation.

Remarque : Cette collection de rubriques suppose que vous avez une connaissance de base de l'architecture SNA (Systems Network Architecture). L'autre documentation mentionnée dans cette section contient une brève introduction aux concepts et à la terminologie pertinents. Si vous avez besoin d'une introduction technique plus complète à SNA, voir *Systems Network Architecture Technical Overview*, GC30-3073.

SNA LU 6.2 fournit trois services de sécurité:

- Cryptographie de niveau session
- Authentification au niveau de la session
- Authentification au niveau de la conversation

Pour la cryptographie au niveau de la session et l'authentification au niveau de la session, SNA utilise l'algorithme *Data Encryption Standard (DES)* . L'algorithme DES est un algorithme de chiffrement par blocs qui utilise une clé symétrique pour le chiffrement et le déchiffrement des données. La longueur du bloc et de la clé est de 8 octets.

Cryptographie de niveau session
La *cryptographie au niveau de la session* chiffre et déchiffre les données de session à l'aide de l'algorithme DES. Il peut donc être utilisé pour fournir un service de confidentialité de niveau liaison sur les canaux SNA LU 6.2 .

Les unités logiques peuvent fournir une cryptographie de données obligatoire (ou obligatoire), une cryptographie de données sélective ou aucune cryptographie de données.

Dans une *session cryptographique obligatoire*, une unité logique chiffre toutes les unités de demande de données sortantes et déchiffre toutes les unités de demande de données entrantes.

Dans une *session cryptographique sélective*, une unité logique chiffre uniquement les unités de demande de données spécifiées par le programme de transaction d'envoi (TP). L'unité logique émettrice signale que les données sont chiffrées en définissant un indicateur dans l'en-tête de demande. En vérifiant cet indicateur, la LU réceptrice peut savoir quelles unités de requête déchiffrer avant de les transmettre à la TP réceptrice.

Dans un réseau SNA, les agents IBM MQ MCA sont des programmes de transaction. Les agents MCA ne demandent pas de chiffrement pour les données qu'ils envoient. La cryptographie sélective de données n'est donc pas une option ; seule la cryptographie de données obligatoire ou aucune cryptographie de données est possible sur une session.

Pour plus d'informations sur l'implémentation de la cryptographie de données obligatoire, voir la documentation de votre sous-système SNA. Consultez la même documentation pour plus d'informations sur les formes de chiffrement plus fortes qui peuvent être disponibles sur votre plateforme, comme le chiffrement Triple DES 24 octets sur z/OS.

Pour plus d'informations sur la cryptographie de niveau session, voir *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808.

Authentification au niveau de la session

L' *authentification au niveau de la session* est un protocole de sécurité au niveau de la session qui permet à deux unités logiques de s'authentifier l'une l'autre lors de l'activation d'une session. Elle est également appelée *vérification LU-LU*.

Etant donné qu'une unité logique est effectivement la "passerelle" dans un système à partir du réseau, vous pouvez considérer que ce niveau d'authentification est suffisant dans certaines circonstances. Par exemple, si votre gestionnaire de files d'attente doit échanger des messages avec un gestionnaire de files d'attente éloignées qui s'exécute dans un environnement contrôlé et sécurisé, vous pouvez être prêt à faire confiance aux identités des autres composants du système distant une fois que l'unité logique a été authentifiée.

L'authentification au niveau de la session est effectuée par chaque unité logique qui vérifie le mot de passe de son partenaire. Le mot de passe est appelé *mot de passe LU-LU* car un mot de passe est établi entre chaque paire d'unités logiques. Le mode d'établissement d'un mot de passe LU-LU dépend de l'implémentation et est hors de la portée de SNA.

La Figure 12, à la page 120 illustre les flux d'authentification au niveau de la session.

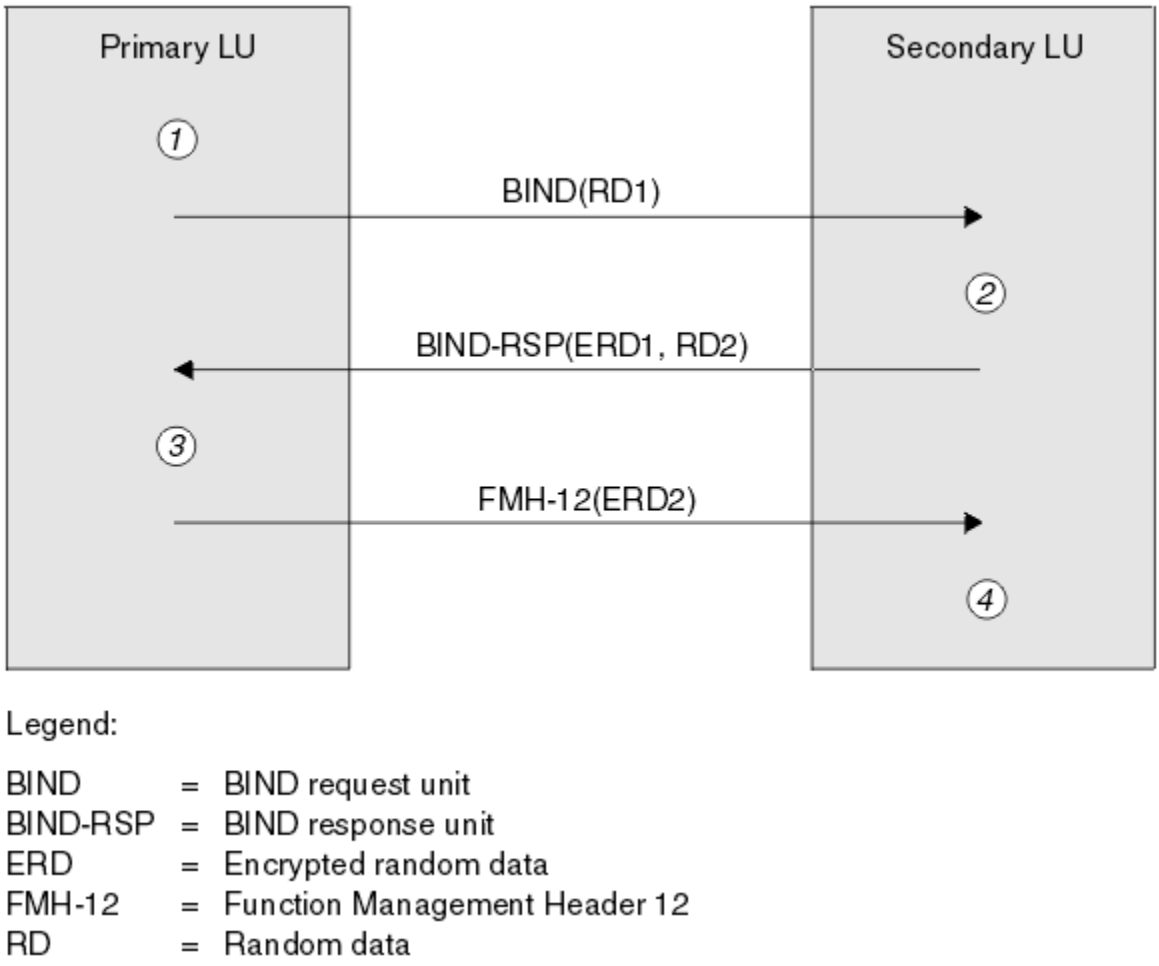


Figure 12. Flux pour l'authentification au niveau de la session

Le protocole d'authentification au niveau de la session est le suivant. Les nombres de la procédure correspondent aux nombres de Figure 12, à la page 120.

1. L'unité logique principale génère une valeur de données aléatoire (RD1) et l'envoie à l'unité logique secondaire dans la demande BIND.
2. Lorsque la LU secondaire reçoit la requête BIND avec les données aléatoires, elle chiffre les données à l'aide de l'algorithme DES avec sa copie du mot de passe LU-LU comme clé. L'unité logique secondaire génère ensuite une deuxième valeur de données aléatoires (RD2) et l'envoie, avec les données chiffrées (ERD1), à l'unité logique principale dans la réponse BIND.
3. Lorsque l'unité logique principale reçoit la réponse BIND, elle calcule sa propre version des données chiffrées à partir des données aléatoires qu'elle a générées à l'origine. Pour ce faire, il utilise l'algorithme DES avec sa copie du mot de passe LU-LU comme clé. Il compare ensuite sa version aux données chiffrées qu'il a reçues dans la réponse BIND. Si les deux valeurs sont identiques, l'unité logique principale sait que l'unité logique secondaire possède le même mot de passe et que l'unité logique secondaire est authentifiée. Si les deux valeurs ne correspondent pas, l'unité logique principale met fin à la session.

   L'unité logique principale chiffre ensuite les données aléatoires qu'elle a reçues dans la réponse BIND et envoie les données chiffrées (ERD2) à l'unité logique secondaire dans un en-tête de gestion de fonction 12 (FMH-12).
4. Lorsque l'unité logique secondaire reçoit le FMH-12, elle calcule sa propre version des données chiffrées à partir des données aléatoires qu'elle a générées. Il compare ensuite sa version aux données chiffrées qu'il a reçues dans le FMH-12. Si les deux valeurs sont identiques, l'unité logique principale est authentifiée. Si les deux valeurs ne correspondent pas, l'unité logique secondaire met fin à la session.

Dans une version améliorée du protocole, qui offre une meilleure protection contre les attaques de l'homme du milieu, la LU secondaire calcule un code d'authentification de message (MAC) DES à partir de RD1, RD2et du nom qualifié complet de la LU secondaire, en utilisant sa copie du mot de passe LU-LU comme clé. L'unité logique secondaire envoie l'adresse MAC à l'unité logique principale dans la réponse BIND au lieu de ERD1.

L'unité logique principale authentifie l'unité logique secondaire en calculant sa propre version du MAC, qu'elle compare au MAC reçu dans la réponse BIND. L'unité logique principale calcule ensuite une seconde adresse MAC à partir de RD1 et RD2, et envoie l'adresse MAC à l'unité logique secondaire dans FMH-12 au lieu de ERD2.

L'unité logique secondaire authentifie l'unité logique principale en calculant sa propre version du deuxième MAC, qu'elle compare avec le MAC reçu dans le FMH-12.

Pour plus d'informations sur la configuration de l'authentification au niveau de la session, voir la documentation de votre sous-système SNA. Pour plus d'informations sur l'authentification de niveau session, voir *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808.

Authentification au niveau de la conversation
Lorsqu'un programme transactionnel local tente d'allouer une conversation avec un programme transactionnel partenaire, l'unité logique locale envoie une demande de connexion à l'unité logique partenaire, en lui demandant de connecter le programme transactionnel partenaire. Dans certaines circonstances, la demande d'association peut contenir des informations de sécurité que l'unité logique partenaire peut utiliser pour authentifier le TP local. Il s'agit de l' *authentification au niveau de la conversation*ou de la *vérification de l'utilisateur final*.

Les rubriques suivantes décrivent comment IBM MQ fournit la prise en charge de l'authentification au niveau de la conversation.

Pour plus d'informations sur l'authentification au niveau de la conversation, voir *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808. Pour des informations spécifiques à z/OS, voir *z/OS MVS Planning: APPC/MVS Management*, SA22-7599.

Pour plus d'informations sur CPI-C, voir *Common Programming Interface Communications CPI-C Specification*, SC31-6180. Pour plus d'informations sur APPC/MVS TP Conversation Callable Services, voir *z/OS MVS Programming: Writing Transaction Programs for APPC/MVS*, SA22-7621.

Windows IBM i UNIX *Prise en charge de l'authentification au niveau de la conversation sur IBM i, UNIXet Windows*

Utilisez cette rubrique pour obtenir une présentation du fonctionnement de l'authentification au niveau de la conversation sur IBM i, UNIXet Windows.

La prise en charge de l'authentification au niveau de la conversation sous IBM i, UNIXet Windows est illustrée dans Figure 13, à la page 122. Les numéros du diagramme correspondent aux numéros de la description qui suit.

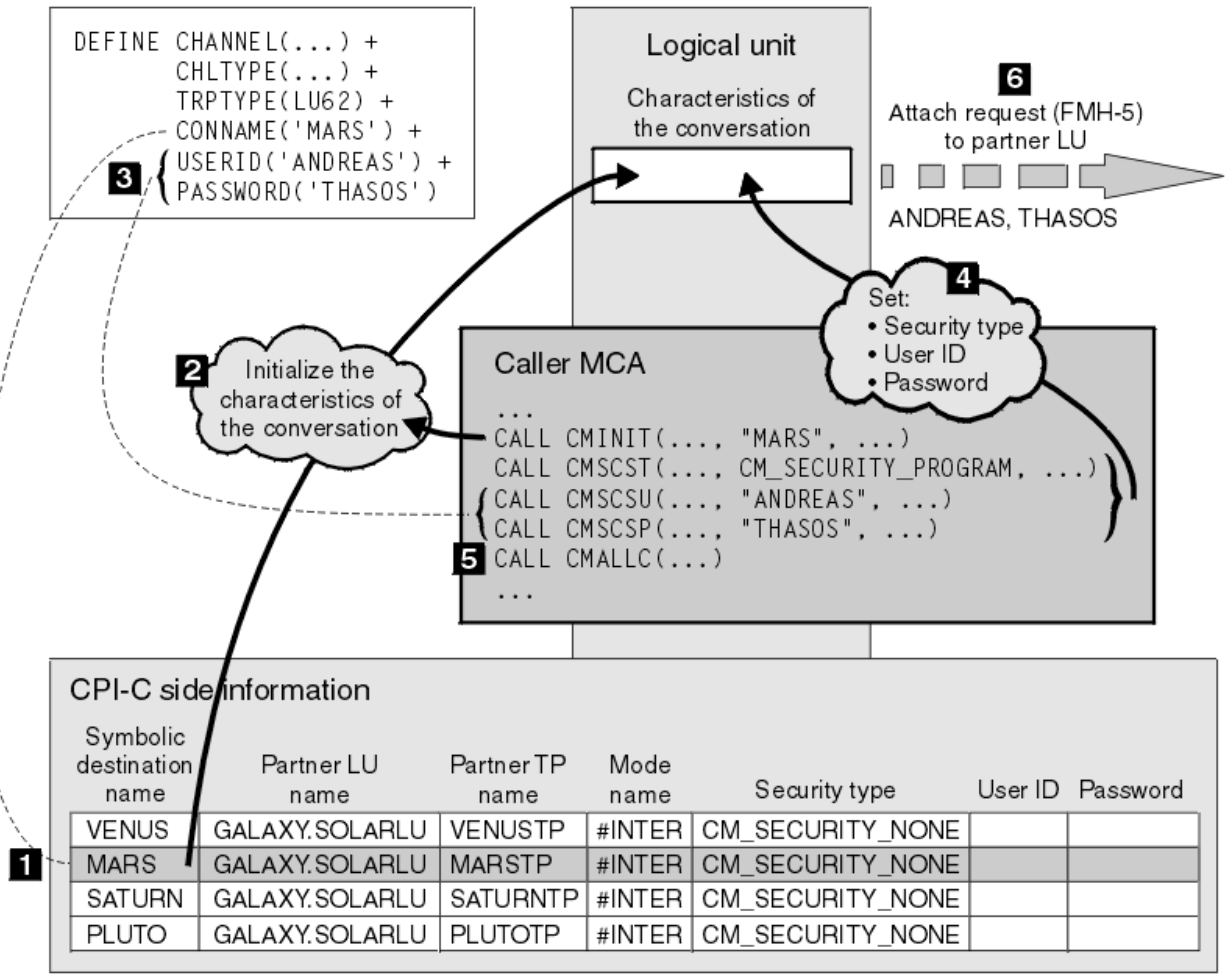


Figure 13. Prise en charge par IBM MQ de l'authentification au niveau de la conversation

Sous IBM i, UNIXet Windows, un agent MCA utilise des appels CPI-C (Common Programming Interface Communications) pour communiquer avec un agent MCA partenaire sur un réseau SNA. Dans la définition de canal à l'extrémité appelante d'un canal, la valeur du paramètre CONNAME est un nom de destination symbolique qui identifie une entrée d'informations côté CPI-C (1). Cette entrée indique:

- Nom de l'unité logique partenaire
- Nom du programme transactionnel partenaire, qui est un agent MCA répondeur
- Nom du mode à utiliser pour la conversation

Une entrée d'informations complémentaires peut également spécifier les informations de sécurité suivantes:

- Type de sécurité.

Les types de sécurité couramment implémentés sont CM_SECURITY_NONE, CM_SECURITY_PROGRAM et CM_SECURITY_SAME, mais d'autres sont définis dans la spécification CPI-C.

- ID utilisateur.
- Un mot de passe.

Un agent MCA appelant se prépare à allouer une conversation avec un agent MCA répondeur en émettant l'appel CPI-C CMINIT, en utilisant la valeur de CONNAME comme l'un des paramètres de l'appel. L'appel CMINIT identifie, pour le bénéfice de l'unité logique locale, l'entrée d'informations complémentaires que l'agent MCA a l'intention d'utiliser pour la conversation. L'unité logique locale utilise les valeurs de cette entrée pour initialiser les caractéristiques de la conversation (2).

L'agent MCA appelant vérifie ensuite les valeurs des paramètres USERID et PASSWORD dans la définition de canal (3). Si USERID est défini, l'agent MCA appelant émet les appels CPI-C suivants (4):

- CMSCST, pour définir le type de sécurité de la conversation sur CM_SECURITY_PROGRAM.
- CMSCSU, pour définir l'ID utilisateur de la conversation sur la valeur USERID.
- CMSCSP, pour définir le mot de passe de la conversation sur la valeur PASSWORD. CMSCSP n'est pas appelé sauf si PASSWORD est défini.

Le type de sécurité, l'ID utilisateur et le mot de passe définis par ces appels remplacent toutes les valeurs précédemment acquises à partir de l'entrée d'informations complémentaires.

L'agent MCA appelant émet ensuite l'appel CPI-C CMALLC pour allouer la conversation (5). En réponse à cet appel, l'unité logique locale envoie une demande d'association (Function Management Header 5, ou FMH-5) à l'unité logique partenaire (6).

Si l'unité logique partenaire accepte un ID utilisateur et un mot de passe, les valeurs USERID et PASSWORD sont incluses dans la demande d'association. Si l'unité logique partenaire n'accepte pas d'ID utilisateur et de mot de passe, les valeurs ne sont pas incluses dans la demande d'association. L'unité logique locale détermine si l'unité logique partenaire accepte un ID utilisateur et un mot de passe dans le cadre d'un échange d'informations lorsque les unités logiques se lient pour former une session.

Dans une version ultérieure de la demande d'association, un remplacement de mot de passe peut se produire entre les unités logiques au lieu d'un mot de passe clair. Un remplaçant de mot de passe est un code d'authentification de message DES (MAC) ou un résumé de message SHA-1 , formé à partir du mot de passe. Les remplacements de mot de passe ne peuvent être utilisés que si les deux unités logiques les prennent en charge.

Lorsque l'unité logique partenaire reçoit une demande d'association entrante contenant un ID utilisateur et un mot de passe, elle peut utiliser l'ID utilisateur et le mot de passe à des fins d'identification et d'authentification. En faisant référence aux listes de contrôle d'accès, l'unité logique partenaire peut également déterminer si l'ID utilisateur a le droit d'allouer une conversation et de connecter l'agent MCA répondeur.

En outre, l'agent MCA répondeur peut s'exécuter sous l'ID utilisateur inclus dans la demande d'association.Dans ce cas, l'ID utilisateur devient l'ID utilisateur par défaut pour l'agent MCA répondeur et est utilisé pour les vérifications des droits d'accès lorsque l'agent MCA tente de se connecter au gestionnaire de files d'attente. Il peut également être utilisé pour les vérifications des droits d'accès ultérieures lorsque l'agent MCA tente d'accéder aux ressources du gestionnaire de files d'attente.

La manière dont un ID utilisateur et un mot de passe dans une demande de connexion peuvent être utilisés pour l'identification, l'authentification et le contrôle d'accès dépend de l'implémentation. Pour des informations spécifiques à votre sous-système SNA, reportez-vous à la documentation appropriée.

Si USERID n'est pas défini, l'agent MCA appelant n'appelle pas CMSCST, CMSCSU et CMSCSP. Dans ce cas, les informations de sécurité qui circulent dans une demande d'association sont uniquement déterminées par ce qui est spécifié dans l'entrée d'informations complémentaires et par ce que l'unité logique partenaire accepte.

Authentification au niveau de la conversation et IBM MQ for z/OS
Utilisez cette rubrique pour obtenir une vue d'ensemble du fonctionnement de l'authentification au niveau de la conversation sur z/OS.

Sous IBM MQ for z/OS, les agents MCA n'utilisent pas CPI-C. Au lieu de cela, ils utilisent APPC/MVS TP Conversation Callable Services, une implémentation de la communication évoluée de programme à programme (APPC), qui dispose de certaines fonctions CPI-C. Lorsqu'un agent MCA appelant alloue une conversation, un type de sécurité SAME est spécifié sur l'appel. Par conséquent, comme une unité logique APPC/MVS prend en charge la vérification permanente uniquement pour les conversations entrantes et non pour les conversations sortantes, il existe deux possibilités:

- Si l'unité logique partenaire fait confiance à l'unité logique APPC/MVS et accepte un ID utilisateur déjà vérifié, l'unité logique APPC/MVS envoie une demande d'association contenant:
  - ID utilisateur de l'espace adresse de l'initiateur de canal
  - Un nom de profil de sécurité, qui, si RACF est utilisé, est le nom du groupe de connexion en cours de l'ID utilisateur de l'espace adresse de l'initiateur de canal
  - Un indicateur déjà vérifié
- Si l'unité logique partenaire ne fait pas confiance à l'unité logique APPC/MVS et n'accepte pas un ID utilisateur déjà vérifié, l'unité logique APPC/MVS envoie une demande d'association ne contenant aucune information de sécurité.

Sous IBM MQ for z/OS, les paramètres USERID et PASSWORD de la commande DEFINE CHANNEL ne peuvent pas être utilisés pour un canal de transmission de messages et sont valides uniquement à l'extrémité de connexion client d'un canal MQI. Par conséquent, une demande d'association d'une unité logique APPC/MVS ne contient jamais de valeurs spécifiées par ces paramètres.

Sécurité des clusters de gestionnaires de files d'attente

Bien que les clusters de gestionnaires de files d'attente soient pratiques à utiliser, vous devez accorder une attention particulière à leur sécurité.

Un *cluster de gestionnaires de files d'attente* est un réseau de gestionnaires de files d'attente associés de manière logique. Un gestionnaire de files d'attente qui est membre d'un cluster est appelé *gestionnaire de files d'attente de cluster*.

Une file d'attente appartenant à un gestionnaire de files d'attente de cluster peut être rendue connue des autres gestionnaires de files d'attente du cluster. Cette file d'attente est appelée *file d'attente de cluster*. Tout gestionnaire de files d'attente d'un cluster peut envoyer des messages à des files d'attente de cluster sans avoir besoin de l'un des éléments suivants:

- Une définition de file d'attente éloignée explicite pour chaque file d'attente de cluster
- Canaux définis explicitement vers et depuis chaque gestionnaire de files d'attente éloignées
- Une file d'attente de transmission distincte pour chaque canal sortant

Vous pouvez créer un cluster dans lequel au moins deux gestionnaires de files d'attente sont des clones. Cela signifie qu'ils possèdent des instances des mêmes files d'attente locales, y compris des files d'attente locales déclarées comme files d'attente de cluster, et qu'ils peuvent prendre en charge des instances des mêmes applications serveur.

Lorsqu'une application connectée à un gestionnaire de files d'attente de cluster envoie un message à une file d'attente de cluster comportant une instance sur chacun des gestionnaires de files d'attente clonés, IBM MQ décide à quel gestionnaire de files d'attente elle doit être envoyée. Lorsque de nombreuses applications envoient des messages à la file d'attente de cluster, IBM MQ équilibre la charge de travail entre chacun des gestionnaires de files d'attente ayant une instance de la file d'attente. Si l'un des systèmes hébergeant un gestionnaire de files d'attente cloné est défaillant, IBM MQ continue d'équilibrer la charge de travail entre les gestionnaires de files d'attente restants jusqu'à ce que le système défaillant soit redémarré.

Si vous utilisez des clusters de gestionnaires de files d'attente, vous devez prendre en compte les problèmes de sécurité suivants:

- Autoriser uniquement les gestionnaires de files d'attente sélectionnés à envoyer des messages à votre gestionnaire de files d'attente

- Autoriser uniquement les utilisateurs sélectionnés d'un gestionnaire de files d'attente éloignées à envoyer des messages à une file d'attente de votre gestionnaire de files d'attente
- Autoriser les applications connectées à votre gestionnaire de files d'attente à envoyer des messages uniquement aux files d'attente éloignées sélectionnées

Ces considérations sont pertinentes même si vous n'utilisez pas de clusters, mais elles deviennent plus importantes si vous utilisez des clusters.

Si une application peut envoyer des messages à une file d'attente de cluster, elle peut envoyer des messages à n'importe quelle autre file d'attente de cluster sans avoir besoin de définitions de file d'attente éloignée, de files d'attente de transmission ou de canaux supplémentaires. Il est donc plus important de déterminer si vous devez restreindre l'accès aux files d'attente de cluster sur votre gestionnaire de files d'attente et de limiter les files d'attente de cluster auxquelles vos applications peuvent envoyer des messages.

Des considérations de sécurité supplémentaires s'appliquent uniquement si vous utilisez des clusters de gestionnaires de files d'attente:

- Autoriser uniquement les gestionnaires de files d'attente sélectionnés à rejoindre un cluster
- Forcer les gestionnaires de files d'attente indésirables à quitter un cluster

Pour plus d'informations sur toutes ces considérations, voir Maintien de la sécurité des clusters.

z/OS Pour des considérations spécifiques à IBM MQ for z/OS, voir «Sécurité dans les clusters de gestionnaires de files d'attente sous z/OS», à la page 271.

Tâches associées

«Empêcher les gestionnaires de files d'attente de recevoir des messages», à la page 477
Vous pouvez empêcher un gestionnaire de files d'attente de cluster de recevoir des messages qu'il n'est pas autorisé à recevoir à l'aide de programmes d'exit.

Sécurité pour la publication / abonnement IBM MQ

Des considérations de sécurité supplémentaires sont à prendre en compte si vous utilisez la fonction de publication / abonnement IBM MQ .

Dans un système de publication / abonnement, il existe deux types d'application: le diffuseur de publications et l'abonné. Les *diffuseurs de publications* fournissent des informations sous la forme de messages IBM MQ . Lorsqu'un diffuseur de publications publie un message, il spécifie une *rubrique*qui identifie l'objet des informations contenues dans le message.

Les *abonnés* sont les consommateurs des informations qui sont publiées. Un abonné spécifie les rubriques qui l'intéressent en s'y abonnant.

Le *gestionnaire de files d'attente* est une application fournie avec IBM MQ Publish / Subscribe. Il reçoit les messages publiés des diffuseurs de publications et les demandes d'abonnement des abonnés, et achemine les messages publiés vers les abonnés. Un abonné reçoit des messages uniquement sur les sujets auxquels il s'est abonné.

Pour plus d'informations, voir Sécurité de publication / abonnement.

Sécurité de multidiffusion

Utilisez ces informations pour comprendre pourquoi des processus de sécurité peuvent être nécessaires avec IBM MQ Multicast.

IBM MQ Multicast ne dispose pas de sécurité intégrée. Les contrôles de sécurité sont gérés dans le gestionnaire de files d'attente au moment de l'opération MQOPEN et le paramètre de zone MQMD est géré par le client. Certaines applications du réseau peuvent ne pas être des applications IBM MQ (par exemple, les applications LLM, voir Interopérabilité multidiffusion avec IBM MQ Low Latency Messaging pour plus d'informations). Par conséquent, vous devrez peut-être implémenter vos propres procédures de sécurité car les applications de réception ne peuvent pas être certaines de la validité des zones de contexte.

Il existe trois processus de sécurité à prendre en compte:

Contrôle d'accès

Le contrôle d'accès dans IBM MQ est basé sur les ID utilisateur. Pour plus d'informations sur ce sujet, voir «Contrôle d'accès pour les clients», à la page 99.

Sécurité réseau

Un réseau isolé peut être une option de sécurité viable pour empêcher les faux messages. Il est possible qu'une application de l'adresse de groupe de multidiffusion publie des messages malveillants à l'aide de fonctions de communication natives, qui ne peuvent pas être distinguées des messages MQ car elles proviennent d'une application de la même adresse de groupe de multidiffusion.

Il est également possible qu'un client sur l'adresse de groupe de multidiffusion reçoive des messages destinés à d'autres clients sur la même adresse de groupe de multidiffusion.

L'isolement du réseau de multidiffusion garantit que seuls les clients et les applications valides y ont accès. Cette précaution de sécurité peut empêcher l'entrée de messages malveillants et la sortie d'informations confidentielles.

Pour plus d'informations sur les adresses réseau de groupe de multidiffusion, voir: Définition du réseau approprié pour le trafic de multidiffusion

Signatures numériques

Une signature numérique est formée par le chiffrement d'une représentation d'un message. Le chiffrement utilise la clé privée du signataire et, pour des raisons d'efficacité, opère généralement sur un résumé de message plutôt que sur le message lui-même. La signature numérique d'un message avant une opération MQPUT est une bonne précaution de sécurité, mais ce processus peut avoir un impact négatif sur les performances s'il existe un volume important de messages.

Les signatures numériques varient en fonction des données en cours de signature. Si deux messages différents sont signés numériquement par la même entité, les deux signatures diffèrent, mais les deux signatures peuvent être vérifiées avec la même clé publique, c'est-à-dire la clé publique de l'entité qui a signé les messages.

Comme indiqué précédemment dans cette section, il est possible qu'une application sur l'adresse de groupe de multidiffusion publie des messages malveillants à l'aide de fonctions de communication natives, qui ne peuvent pas être distinguées des messages MQ . Les signatures numériques fournissent une preuve de l'origine, et seul l'expéditeur connaît la clé privée, ce qui fournit des preuves solides que l'expéditeur est l'émetteur du message.

Pour plus d'informations sur ce sujet, voir «Concepts cryptographiques», à la page 7.

Pare-feux et passe-système Internet

Normalement, vous utilisez un pare-feu pour empêcher l'accès à partir d'adresses IP hostiles, par exemple lors d'une attaque par refus de service. Toutefois, il peut être nécessaire de bloquer temporairement les adresses IP dans IBM MQ, peut-être pendant que vous attendez qu'un administrateur de sécurité mette à jour les règles de pare-feu.

Pour bloquer une ou plusieurs adresses IP, créez un enregistrement d'authentification de canal de type BLOCKADDR ou ADDRESSMAP. Pour plus d'informations, voir «Blocage d'adresses IP spécifiques», à la page 397.

Sécurité de IBM MQ Internet Pass-Thru

IBM MQ Internet Pass-Thru peut simplifier la communication via un pare-feu, mais cela a des implications sur la sécurité.

IBM MQ Internet Pass-Thru (MQIPT) est un composant facultatif d'IBM MQ que vous pouvez utiliser pour implémenter des solutions de messagerie entre des sites distants sur Internet.

MQIPT permet à deux gestionnaires de files d'attente d'échanger des messages ou à une application client IBM MQ de se connecter à un gestionnaire de files d'attente via Internet sans nécessiter de

connexion TCP/IP directe. Cela est utile si un pare-feu interdit une connexion TCP/IP directe entre deux systèmes. Il rend le passage des flux de protocole de canal IBM MQ vers et depuis un pare-feu plus simple et plus facile à gérer en tunnelisant les flux dans HTTP ou en agissant en tant que proxy. A l'aide du protocole TLS (Transport Layer Security), il peut également être utilisé pour chiffrer et déchiffrer les messages envoyés sur Internet.

Lorsque votre système IBM MQ communique avec MQIPT, à moins que vous n'utilisiez le mode proxy SSL dans MQIPT, vérifiez que la propriété CipherSpec utilisée par IBM MQ correspond à la propriété CipherSuite utilisée par MQIPT:

- Lorsque MQIPT agit en tant que serveur TLS et que IBM MQ se connecte en tant que client TLS, le CipherSpec utilisé par IBM MQ doit correspondre à une CipherSuite activée dans le fichier de clés MQIPT approprié.
- Lorsque MQIPT agit en tant que client TLS et se connecte à un serveur TLS IBM MQ , la suite de chiffrement MQIPT CipherSuite doit correspondre à la spécification de chiffrement CipherSpec définie sur le canal IBM MQ récepteur.

Si vous migrez depuis MQIPT vers la prise en charge TLS IBM MQ intégrée, transférez les certificats numériques depuis le fichier de clés MQIPT à l'aide de **mqiptKeyman** ou de **mqiptKeycmd**.

Pour plus d'informations, voir IBM MQ Internet Pass-Thru.

z/OS Liste de contrôle d'implémentation de la sécurité IBM MQ for z/OS

Cette rubrique fournit une procédure pas à pas que vous pouvez utiliser pour déterminer et définir l'implémentation de la sécurité pour chacun de vos gestionnaires de files d'attente IBM MQ .

RACF fournit des définitions pour les classes de sécurité IBM MQ dans sa table CDT (Class Descriptor Table) statique fournie. Au fur et à mesure que vous utilisez la liste de contrôle, vous pouvez déterminer les classes requises par votre configuration. Vous devez vous assurer qu'ils sont activés comme décrit dans «Classes de sécurité RACF», à la page 187.

Pour plus de détails, voir les autres sections, en particulier «Profils utilisés pour contrôler l'accès aux ressources IBM MQ», à la page 198.

Si vous avez besoin d'un contrôle de sécurité, suivez la liste de contrôle suivante pour l'implémenter:

1. Activez la classe RACF MQADMIN (profils en majuscules) ou MXADMIN (profils en casse mixte).
   - Voulez-vous la sécurité au niveau du groupe de partage de files d'attente, au niveau du gestionnaire de files d'attente ou une combinaison des deux?

     Voir «Profils permettant de contrôler la sécurité au niveau du groupe de partage de files d'attente ou du gestionnaire de files d'attente», à la page 193.
2. Avez-vous besoin de la sécurité de la connexion?
   - **Oui**: activez la classe MQCONN. Définissez les profils de connexion appropriés au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQCONN. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.

     Remarque : Seuls les utilisateurs de la demande d'API `MQCONN` ou les ID utilisateur de l'espace adresse CICS ou IMS doivent avoir accès au profil de connexion correspondant.
   - **Non**: définissez un hlq.NO.CONNECT.CHECKS au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQADMIN ou MXADMIN.
3. Avez-vous besoin d'un contrôle de sécurité sur les commandes?
   - **Oui**: activez la classe MQCMDS. Définissez les profils de commande appropriés au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQCMDS. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.

     Si vous utilisez un groupe de partage de files d'attente, vous devrez peut-être inclure les ID utilisateur utilisés par le gestionnaire de files d'attente lui-même et l'initiateur de canal. Voir «Configuration de la sécurité des ressources IBM MQ for z/OS», à la page 262.

- **Non**: définissez un hlq.NO.CMD.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.

4. Avez-vous besoin de sécurité sur les ressources utilisées dans les commandes?
   - **Oui**: Vérifiez que la classe MQADMIN ou MXADMIN est active. Définissez les profils appropriés pour la protection des ressources sur les commandes au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQADMIN ou MXADMIN. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils. Définissez le paramètre CMDUSER dans CSQ6SYSP sur l'ID utilisateur par défaut à utiliser pour les contrôles de sécurité des commandes.

     Si vous utilisez un groupe de partage de files d'attente, vous devrez peut-être inclure les ID utilisateur utilisés par le gestionnaire de files d'attente lui-même et l'initiateur de canal. Voir «Configuration de la sécurité des ressources IBM MQ for z/OS», à la page 262.
   - **Non**: définissez un hlq.NO.CMD.RESC.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.
5. Avez-vous besoin de la sécurité de la file d'attente?
   - **Oui**: Activez la classe MQQUEUE ou MXQUEUE. Définissez les profils de file d'attente appropriés pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans MQQUEUE ou MXQUEUEclass. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.
   - **Non**: définissez un hlq.NO.QUEUE.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.
6. Avez-vous besoin de la sécurité des processus?
   - **Oui**: Activez la classe MQPROC ou MXPROC. Définissez les profils de processus appropriés au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente et autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.
   - **Non**: définissez un hlq.NO.PROCESS.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente approprié dans la classe MQADMIN ou MXADMIN.
7. Avez-vous besoin de la sécurité de la liste de noms?
   - **Oui**: Activez MQNLIST ou MXNLISTclass. Définissez les profils de liste de noms appropriés au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQNLIST ou MXNLIST. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.
   - **Non**: définissez un hlq.NO.NLIST.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.
8. Avez-vous besoin de la sécurité des rubriques?
   - **Oui**: activez la classe MXTOPIC. Définissez les profils de rubrique appropriés au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MXTOPIC. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.
   - **Non**: définissez un hlq.NO.TOPIC.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.
9. Les utilisateurs ont-ils besoin de protéger l'utilisation des options `MQOPEN` ou `MQPUT1` relatives à l'utilisation du contexte?
   - **Oui**: Vérifiez que la classe MQADMIN ou MXADMIN est active. Définissez les profils hlq.CONTEXT.queuename au niveau de la file d'attente, du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQADMIN ou MXADMIN. Ensuite, autorisez les utilisateurs ou les groupes appropriés à accéder à ces profils.
   - **Non**: définissez un hlq.NO.CONTEXT.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.
10. Avez-vous besoin de protéger l'utilisation d'autres ID utilisateur?

- **Oui**: Vérifiez que la classe MQADMIN ou MXADMIN est active. Définissez le fichier hlq.ALTERNATE.USER. Les profils *alternateuserid* pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis et permettent aux utilisateurs ou aux groupes requis d'accéder à ces profils.
- **Non**: Définissez le profil hlq.NO.ALTERNATE.USER.CHECKS pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis dans la classe MQADMIN ou MXADMIN.

11. Avez-vous besoin de personnaliser les ID utilisateur à utiliser pour les contrôles de sécurité des ressources via RESLEVEL?
    - **Oui**: Vérifiez que la classe MQADMIN ou MXADMIN est active. Définissez un profil hlq.RESLEVEL au niveau du gestionnaire de files d'attente ou du groupe de partage de files d'attente dans la classe MQADMIN ou MXADMIN. Ensuite, autorisez les utilisateurs ou les groupes requis à accéder au profil.
    - **Non**: Vérifiez qu'il n'existe aucun profil générique dans la classe MQADMIN ou MXADMIN pouvant s'appliquer à hlq.RESLEVEL. Définissez un profil hlq.RESLEVEL pour le gestionnaire de files d'attente ou le groupe de partage de files d'attente requis et vérifiez qu'aucun utilisateur ou groupe n'y a accès.
12. Avez-vous besoin de'délai d'attente'pour les ID utilisateur inutilisés de IBM MQ ?
    - **Oui**: Déterminez les valeurs de délai d'attente que vous souhaitez utiliser et émettez la commande MQSC ALTER SECURITY pour modifier les paramètres TIMEOUT et INTERVAL.
    - **Non**: Exécutez la commande MQSC ALTER SECURITY pour définir la valeur INTERVAL sur zéro.

    Remarque : Mettez à jour le fichier d'entrée d'initialisation CSQINP1 utilisé par votre sous-système de sorte que la commande MQSC ALTER SECURITY soit émise automatiquement lorsque le gestionnaire de files d'attente est démarré.
13. Utilisez-vous la mise en file d'attente répartie?
    - **Oui**: Utilisez les enregistrements d'authentification de canal. Pour plus d'informations, voir «Enregistrements d'authentification de canal», à la page 50.
    - Vous pouvez également déterminer la valeur d'attribut MCAUSER appropriée pour chaque canal ou fournir des exits de sécurité de canal appropriés.
14. Voulez-vous utiliser TLS (Transport Layer Security)?
    - **Oui**: Pour indiquer que tout utilisateur présentant un certificat personnel TLS contenant un nom distinctif spécifié doit utiliser un utilisateur MCAUSER spécifique, définissez un enregistrement d'authentification de canal de type SSLPEERMAP. Vous pouvez indiquer un nom distinctif unique ou un modèle comportant des caractères génériques.
    - Planifiez votre infrastructure TLS. Installez la fonction System SSL de z/OS. Dans RACF, configurez vos filtres de nom de certificat (CNF), si vous les utilisez, ainsi que vos certificats numériques. Configurez votre fichier de clés SSL. Vérifiez que l'attribut de gestionnaire de files d'attente SSLKEYR n'est pas vide et qu'il pointe vers votre fichier de clés SSL. Vérifiez également que la valeur de SSLTASKS est au moins égale à 2.
    - **Non**: Vérifiez que SSLKEYR est vide et que SSLTASKS est à zéro.

    Pour plus de détails sur TLS, voir «Protocoles de sécurité TLS dans IBM MQ», à la page 24.
15. Utilisez-vous des clients?
    - **Oui**: Utilisez les enregistrements d'authentification de canal.
    - Vous pouvez également déterminer la valeur d'attribut MCAUSER appropriée pour chaque canal de connexion serveur ou fournir des exits de sécurité de canal appropriés si nécessaire.
16. Vérifiez les paramètres de votre commutateur.

    IBM MQ émet des messages lorsque le gestionnaire de files d'attente est démarré qui affichent vos paramètres de sécurité. Utilisez ces messages pour déterminer si vos commutateurs sont définis correctement.
17. Envoyez-vous des mots de passe à partir d'applications client?

- **Oui**: Vérifiez que la fonction z/OS est installée et que la fonction ICSF (Integrated Cryptographic Service Facility) est démarrée pour une protection optimale.
- **Non**: Vous pouvez ignorer le message d'erreur signalant qu'ICSF n'a pas démarré.

Pour plus d'informations sur ICSF, voir «Utilisation de la fonction ICSF (Integrated Cryptographic Service Facility)», à la page 271

Configuration de la sécurité

Cette collection de rubriques contient des informations spécifiques aux différents systèmes d'exploitation et à l'utilisation des clients.

ULW Configuration de la sécurité sous UNIX, Linux, and Windows

Considérations de sécurité spécifiques aux systèmes UNIX, Linux, and Windows .

Les gestionnaires de files d'attente IBM MQ transfèrent des informations potentiellement utiles. Vous devez donc utiliser un système de droits d'accès pour vous assurer que les utilisateurs non autorisés ne peuvent pas accéder à vos gestionnaires de files d'attente. Prenez en compte les types de contrôle de sécurité suivants:

Qui peut administrer IBM MQ
Vous pouvez définir l'ensemble des utilisateurs qui peuvent émettre des commandes pour administrer IBM MQ.

Qui peut utiliser les objets IBM MQ
Vous pouvez définir les utilisateurs (généralement des applications) qui peuvent utiliser des appels MQI et des commandes PCF pour effectuer les opérations suivantes:

- Qui peut se connecter à un gestionnaire de files d'attente.
- Qui peut accéder aux objets (files d'attente, définitions de processus, listes de noms, canaux, canaux de connexion client, programmes d'écoute, services et objets d'informations d'authentification) et quel type d'accès ils ont à ces objets.
- Qui peut accéder aux messages IBM MQ .
- Qui peut accéder aux informations de contexte associées à un message.

Sécurité des canaux
Vous devez vous assurer que les canaux utilisés pour envoyer des messages aux systèmes distants peuvent accéder aux ressources requises.

Vous pouvez utiliser les fonctions d'exploitation standard pour accorder l'accès aux bibliothèques de programmes, aux bibliothèques de liens MQI et aux commandes. Toutefois, le répertoire contenant les files d'attente et les autres données du gestionnaire de files d'attente est privé pour IBM MQ; n'utilisez pas les commandes du système d'exploitation standard pour accorder ou révoquer des autorisations sur les ressources MQI.

ULW Fonctionnement des autorisations sur UNIX, Linux, and Windows

Les tables de spécification d'autorisation dans les rubriques de cette section définissent précisément le fonctionnement des autorisations et les restrictions qui s'appliquent.

Les tableaux s'appliquent aux situations suivantes:

- Applications qui émettent des appels MQI
- Programmes d'administration qui émettent des commandes MQSC sous forme de fichiers PCF d'échappement
- Programmes d'administration qui émettent des commandes PCF

Dans cette section, les informations sont présentées sous la forme d'un ensemble de tables qui spécifient les éléments suivants:

Action à exécuter
Option MQI, commande MQSC ou commande PCF.

Objet de contrôle d'accès
File d'attente, processus, gestionnaire de files d'attente, liste de noms, informations d'authentification, canal, canal de connexion client, programme d'écoute ou service.

Autorisation requise
Exprimée sous la forme d'une constante MQZAO_.

Dans les tableaux, les constantes préfixées par MQZAO_ correspondent aux mots clés de la liste d'autorisation de la commande `setmqaut` pour l'entité particulière. Par exemple, MQZAO_BROWSE correspond au mot clé `+browse`, MQZAO_SET_ALL_CONTEXT correspond au mot clé `+setall`, etc. Ces constantes sont définies dans le fichier d'en-tête cmqzc.h, fourni avec le produit.

ULW *Autorisations pour les appels MQI*

MQCONN, **MQOPEN**, **MQPUT1**et **MQCLOSE** peuvent nécessiter des vérifications d'autorisation. Les tableaux de cette rubrique récapitulent les autorisations requises pour chaque appel.

Une application est autorisée à émettre des appels et des options MQI spécifiques uniquement si l'identificateur utilisateur sous lequel elle s'exécute (ou dont elle peut assumer les autorisations) a reçu l'autorisation appropriée.

Quatre appels MQI peuvent nécessiter des vérifications d'autorisation: **MQCONN**, **MQOPEN**, **MQPUT1**et **MQCLOSE**.

Pour **MQOPEN** et **MQPUT1**, la vérification des droits d'accès est effectuée sur le nom de l'objet en cours d'ouverture et non sur le ou les noms, ce qui se produit après la résolution d'un nom. Par exemple, une application peut être autorisée à ouvrir une file d'attente alias sans avoir le droit d'ouvrir la file d'attente de base dans laquelle l'alias est résolu. La règle est que la vérification est effectuée sur la première définition rencontrée lors du processus de résolution d'un nom qui n'est pas un alias de gestionnaire de files d'attente, sauf si la définition d'alias de gestionnaire de files d'attente est ouverte directement ; c'est-à-dire que son nom est affiché dans la zone *ObjectName* du descripteur d'objet. Des droits sont toujours nécessaires pour l'objet en cours d'ouverture. Dans certains cas, des droits d'accès supplémentaires indépendants de la file d'attente, obtenus via une autorisation pour l'objet gestionnaire de files d'attente, sont requis.

Tableau 10, à la page 131, Tableau 11, à la page 132, Tableau 12, à la page 132et Tableau 13, à la page 133 récapitulent les autorisations requises pour chaque appel. Dans les tableaux *Non applicable* , le contrôle d'autorisation n'est pas pertinent pour cette opération ; *Pas de contrôle* signifie qu'aucun contrôle d'autorisation n'est effectué.

Remarque : Vous ne trouverez aucune mention des listes de noms, des canaux, des canaux de connexion client, des programmes d'écoute, des services ou des objets d'informations d'authentification dans ces tables. En effet, aucune des autorisations ne s'applique à ces objets, à l'exception de MQOO_INQUIRE, pour lequel les mêmes autorisations s'appliquent que pour les autres objets.

L'autorisation spéciale MQZAO_ALL_MQI inclut toutes les autorisations dans les tables qui sont pertinentes pour le type d'objet, à l'exception de MQZAO_DELETE et MQZAO_DISPLAY, qui sont classées comme autorisations d'administration.

Pour modifier l'une des options de contexte de message, vous devez disposer des autorisations appropriées pour émettre l'appel. Par exemple, pour utiliser MQOO_SET_IDENTITY_CONTEXT ou MQPMO_SET_IDENTITY_CONTEXT, vous devez disposer du droit `+setid` .

Tableau 10. Autorisation de sécurité requise pour les appels MQCONN

Autorisation requise pour:	**Objet de file d'attente («1», à la page 133)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQCONN	Non applicable	Non applicable	MQZAO_CONNECT

Tableau 11. Autorisation de sécurité requise pour les appels MQOPEN

Autorisation requise pour:	Objet de file d'attente («1», à la page 133)	Objet Processus	Objet gestionnaire de files d'attente
MQOO_INTERROGATION	MQZAO_INQUIRE	MQZAO_INQUIRE	MQZAO_INQUIRE
MQOO_BROWSE	MQZAO_PARCOURIR	Non applicable	Aucun contrôle
MQOO_ENTRÉE_ *	MQZAO_ENTREE	Non applicable	Aucun contrôle
MQOO_SAVE_ ALL_CONTEXT («2», à la page 133)	MQZAO_ENTREE	Non applicable	Non applicable
MQOO_OUTPUT (file d'attente normale) («3», à la page 133)	MQZAO_OUTPUT	Non applicable	Non applicable
MQOO_PASS_IDENTITY_CONTEXT («4», à la page 133)	MQZAO_PASS_ IDENTITY_CONTEXT	Non applicable	Aucun contrôle
MQOO_PASS_ALL_ CONTEXT («4», à la page 133, «5», à la page 133)	MQZAO_PASS _ALL_CONTEXT	Non applicable	Aucun contrôle
MQOO_SET_IDENTITY_CONTEXT («4», à la page 133, «5», à la page 133)	MQZAO_SET_IDENTITY_CONTEXT	Non applicable	MQZAO_SET_ IDENTITY_CONTEXT («6», à la page 133)
MQOO_SET_ ALL_CONTEXT («4», à la page 133, «7», à la page 133)	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ ALL_CONTEXT («6», à la page 133)
MQOO_OUTPUT (file d'attente de transmission) («8», à la page 133)	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ ALL_CONTEXT («6», à la page 133)
MQOO_SET	MQZAO_SET	Non applicable	Aucun contrôle
MQOO_ALTERNATE_ AUTORITE UTILISATEUR	(«9», à la page 133)	(«9», à la page 133)	MQZAO_ALTERNATE_ USER_AUTHORITY («9», à la page 133, «10», à la page 133)

Tableau 12. Autorisation de sécurité requise pour les appels MQPUT1

Autorisation requise pour:	Objet de file d'attente («1», à la page 133)	Objet Processus	Objet gestionnaire de files d'attente
MQPMO_PASS_ IDENTITY_CONTEXT	MQZAO_PASS_ IDENTITY_CONTEXT («11», à la page 134)	Non applicable	Aucun contrôle
MQPMO_PASS_ALL _CONTEXT	MQZAO_PASS_ ALL_CONTEXT («11», à la page 134)	Non applicable	Aucun contrôle
MQPMO_SET_IDENTITY_CONTEXT	MQZAO_SET_ IDENTITY_CONTEXT («11», à la page 134)	Non applicable	MQZAO_SET_ IDENTITY_CONTEXT («6», à la page 133)

Tableau 12. Autorisation de sécurité requise pour les appels MQPUT1 (suite)			
Autorisation requise pour:	**Objet de file d'attente («1», à la page 133)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQPMO_SET_ALL_CONTEXT	MQZAO_SET_ALL_CONTEXT **(«11», à la page 134)**	Non applicable	MQZAO_SET_ALL_CONTEXT **(«6», à la page 133)**
(File d'attente de transmission) **(«8», à la page 133)**	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ALL_CONTEXT **(«6», à la page 133)**
MQPMO_ALTERNATE_USER_AUTHORITY	**(«12», à la page 134)**	Non applicable	MQZAO_ALTERNATE_USER_AUTHORITY **(«10», à la page 133)**

Tableau 13. Autorisation de sécurité requise pour les appels MQCLOSE			
Autorisation requise pour:	**Objet de file d'attente («1», à la page 133)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQCO_DELETE	MQZAO_DELETE **(«13», à la page 134)**	Non applicable	Non applicable
MQCO_DELETE _PURGE	MQZAO_DELETE **(«13», à la page 134)**	Non applicable	Non applicable

Remarques relatives aux tableaux:

1. Si vous ouvrez une file d'attente modèle:
   - Le droit MQZAO_DISPLAY est requis pour la file d'attente modèle, en plus du droit d'ouverture de la file d'attente modèle pour le type d'accès pour lequel vous l'ouvrez.
   - Les droits MQZAO_CREATE ne sont pas nécessaires pour créer la file d'attente dynamique.
   - L'ID utilisateur utilisé pour ouvrir la file d'attente modèle reçoit automatiquement tous les droits spécifiques à la file d'attente (équivalents à MQZAO_ALL) pour la file d'attente dynamique créée.
2. MQOO_INPUT_ * doit également être spécifié. Valide pour une file d'attente locale, modèle ou alias.
3. Cette vérification est effectuée pour tous les cas de sortie, à l'exception des files d'attente de transmission (voir la remarque «8», à la page 133).
4. MQOO_OUTPUT doit également être spécifié.
5. MQOO_PASS_IDENTITY_CONTEXT est également impliqué par cette option.
6. Ce droit est requis pour l'objet gestionnaire de files d'attente et la file d'attente particulière.
7. MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT et MQOO_SET_IDENTITY_CONTEXT sont également impliquées par cette option.
8. Cette vérification est effectuée pour une file d'attente locale ou modèle dont l'attribut de file d'attente *Utilisation* est MQUS_TRANSMISSION et qui est ouverte directement pour la sortie. Elle ne s'applique pas si une file d'attente éloignée est ouverte (soit en spécifiant les noms du gestionnaire de files d'attente éloignées et de la file d'attente éloignée, soit en indiquant le nom d'une définition locale de la file d'attente éloignée).
9. Au moins l'une des options MQOO_INQUIRE (pour tout type d'objet) ou MQOO_BROWSE, MQOO_INPUT_ *, MQOO_OUTPUT ou MQOO_SET (pour les files d'attente) doit également être spécifiée. La vérification effectuée est la même que pour les autres options spécifiées, à l'aide de l'identificateur d'utilisateur de remplacement fourni pour les droits sur les objets nommés spécifiques, et des droits d'application en cours pour la vérification MQZAO_ALTERNATE_USER_IDENTIFIER.
10. Cette autorisation permet de spécifier tout ID *AlternateUser* .

11. Une vérification MQZAO_OUTPUT est également effectuée si la file d'attente ne possède pas l'attribut de file d'attente *Usage* MQUS_TRANSMISSION.
12. La vérification effectuée est la même que pour les autres options spécifiées, à l'aide de l'identificateur d'utilisateur de remplacement fourni pour le droit de file d'attente nommé spécifique et du droit d'application en cours pour la vérification MQZAO_ALTERNATE_USER_IDENTIFIER.
13. La vérification est effectuée uniquement si les deux affirmations suivantes sont vraies:
    - Une file d'attente dynamique permanente est en cours de fermeture et de suppression.
    - La file d'attente n'a pas été créée par l'appel MQOPEN qui a renvoyé le descripteur d'objet utilisé.

    Sinon, il n'y a pas de contrôle.

ULW *Autorisations pour les commandes MQSC dans les fichiers PCF d'échappement*

Ces informations récapitulent les autorisations requises pour chaque commande MQSC contenue dans Escape PCF.

Non applicable signifie que cette opération n'est pas pertinente pour ce type d'objet.

L'ID utilisateur sous lequel le programme qui soumet la commande s'exécute doit également disposer des droits suivants:

- Droits MQZAO_CONNECT sur le gestionnaire de files d'attente
- Droit MQZAO_DISPLAY sur le gestionnaire de files d'attente afin d'exécuter des commandes PCF
- Droit d'émettre la commande MQSC dans le texte de la commande Escape PCF

ALTER *objet*

Objet	**Autorisation requise**
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE
Information de communication	MODIFICATION MQZAO_DE

CLEAR *objet*

Objet	**Autorisation requise**
File d'attente	MQZAO_CLEAR
Topic	MQZAO_CLEAR
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable

Objet	Autorisation requise
Informations d'authentification	Non applicable
Canal	Non applicable
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable
Information de communication	Non applicable

DEFINE *objet* NOREPLACE («1», à la page 139)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(«2», à la page 139)**
Topic	MQZAO_CREATE **(«2», à la page 139)**
Processus	MQZAO_CREATE **(«2», à la page 139)**
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_CREATE **(«2», à la page 139)**
Informations d'authentification	MQZAO_CREATE **(«2», à la page 139)**
Canal	MQZAO_CREATE **(«2», à la page 139)**
Canal de connexion client	MQZAO_CREATE **(«2», à la page 139)**
Programme d'écoute	MQZAO_CREATE **(«2», à la page 139)**
Service	MQZAO_CREATE **(«2», à la page 139)**
Information de communication	MQZAO_CREATE **(«2», à la page 139)**

DEFINE *objet* REPLACE («1», à la page 139, «3», à la page 139)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE
Information de communication	MODIFICATION MQZAO_DE

DELETE *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DELETE
Topic	MQZAO_DELETE
Processus	MQZAO_DELETE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_DELETE
Informations d'authentification	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de connexion client	MQZAO_DELETE
Programme d'écoute	MQZAO_DELETE
Service	MQZAO_DELETE
Information de communication	MQZAO_DELETE

DISPLAY *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DISPLAY
Topic	MQZAO_DISPLAY
Processus	MQZAO_DISPLAY
Gestionnaire de files d'attente	MQZAO_DISPLAY
Liste de noms	MQZAO_DISPLAY
Informations d'authentification	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de connexion client	MQZAO_DISPLAY
Programme d'écoute	MQZAO_DISPLAY
Service	MQZAO_DISPLAY
Information de communication	MQZAO_DISPLAY

START *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ

Objet	Autorisation requise
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ
Information de communication	Non applicable

STOP *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ
Information de communication	Non applicable

Commandes relatives aux canaux

Commande	Objet	Autorisation requise
Envoyer une commande PING à un canal	Canal	CONTROLE MQZ
Réinitialisation du canal	Canal	MQZAO_CONTRÔLE_ÉTENDU
Résolution du canal	Canal	MQZAO_CONTRÔLE_ÉTENDU

Commandes d'abonnement

Commande	Objet	Autorisation requise
ALTER SUB	Topic	CONTROLE MQZ
DEFINE SUB	Topic	CONTROLE MQZ
SUPPRIMER DES SOUS	Topic	CONTROLE MQZ
DISPLAY SUB	Topic	MQZAO_DISPLAY

Commandes de sécurité

Commande	Objet	Autorisation requise
SET AUTHREC	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
DELETE AUTHREC	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Paramètre DISPLAY AUTHREC	Gestionnaire de files d'attente	MQZAO_DISPLAY

Commande	Objet	Autorisation requise
DISPLAY AUTHSERV	Gestionnaire de files d'attente	MQZAO_DISPLAY
AFFICHER ENTAUTH	Gestionnaire de files d'attente	MQZAO_DISPLAY
SET CHLAUTH	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
AFFICHER CHLAUTH	Gestionnaire de files d'attente	MQZAO_DISPLAY
REFRESH SECURITY	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE

Affichages de statut

Commande	Objet	Autorisation requise
DISPLAY CHSTATUS	Gestionnaire de files d'attente	MQZAO_DISPLAY Notez que les droits `+inq` (ou de manière équivalente MQZAO_INQUIRE) sont requis sur la file d'attente de transmission si le type de canal est CLUSSDR.
DISPLAY LSSTATUS	Gestionnaire de files d'attente	MQZAO_DISPLAY
AFFICHAGE DE PUBSUB	Gestionnaire de files d'attente	MQZAO_DISPLAY
STATUT DU JEU DE CARACTÈRES D'AFFICHAGE	Gestionnaire de files d'attente	MQZAO_DISPLAY
STATUT DE L'AFFICHAGE	Gestionnaire de files d'attente	MQZAO_DISPLAY
DISPLAY TPSTATUS	Gestionnaire de files d'attente	MQZAO_DISPLAY

Commandes relatives aux clusters

Commande	Objet	Autorisation requise
DISPLAY CLUSQMGR	Gestionnaire de files d'attente	MQZAO_DISPLAY
Actualiser le cluster	Appartenance au groupe'mqm'requise	
Réinitialisation d'un cluster	Appartenance au groupe'mqm'requise	
SUSPEND QMGR	Appartenance au groupe'mqm'requise	
RESUME QMGR	Appartenance au groupe'mqm'requise	

Autres commandes d'administration

Commande	Objet	Autorisation requise
PING QMGR	Gestionnaire de files d'attente	MQZAO_DISPLAY
ACTUALISEZ LE GESTIONNAIRE DE FILES D'ATTENTE	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
RESET QMGR	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
DISPLAY CONN	Gestionnaire de files d'attente	MQZAO_DISPLAY
ARRETER CONN	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE

Remarque :

1. Pour les commandes DEFINE, le droit MQZAO_DISPLAY est également requis pour l'objet LIKE si un tel droit est spécifié, ou sur le système SYSTEM.DEFAULT.xxx si LIKE est omis.
2. Les droits MQZAO_CREATE ne sont pas spécifiques à un objet ou à un type d'objet particulier. Le droit de création est accordé pour tous les objets d'un gestionnaire de files d'attente spécifié, en spécifiant un type d'objet QMGR dans la commande `setmqaut` .
3. Ceci s'applique si l'objet à remplacer existe déjà. Si tel n'est pas le cas, la vérification est celle de l'objet DEFINE NOREPLACE.

Information associée

Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER

ULW *Autorisations pour les commandes PCF*

Cette section récapitule les autorisations requises pour chaque commande PCF.

Aucune vérification signifie qu'aucune vérification d'autorisation n'est effectuée ; *Non applicable* signifie que cette opération n'est pas pertinente pour ce type d'objet.

L'ID utilisateur sous lequel le programme qui soumet la commande s'exécute doit également disposer des droits suivants:

- Droits MQZAO_CONNECT sur le gestionnaire de files d'attente
- Droit MQZAO_DISPLAY sur le gestionnaire de files d'attente afin d'exécuter des commandes PCF

L'autorisation spéciale MQZAO_ALL_ADMIN inclut toutes les autorisations de la liste suivante qui sont pertinentes pour le type d'objet, à l'exception de MQZAO_CREATE, qui n'est pas spécifique à un objet ou à un type d'objet particulier.

Modifier *objet*

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Rubrique	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE
Information de communication	MODIFICATION MQZAO_DE

Effacer *objet*

Objet	Autorisation requise
File d'attente	MQZAO_CLEAR
Rubrique	MQZAO_CLEAR
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable

Objet	Autorisation requise
Informations d'authentification	Non applicable
Canal	Non applicable
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable
Information de communication	Non applicable

Copier *objet* (sans remplacement) (1)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(2)**
Rubrique	MQZAO_CREATE **(2)**
Processus	MQZAO_CREATE **(2)**
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_CREATE **(2)**
Informations d'authentification	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**
Canal de connexion client	MQZAO_CREATE **(2)**
Programme d'écoute	MQZAO_CREATE **(2)**
Service	MQZAO_CREATE **(2)**
Information de communication	MQZAO_CREATE **(«2», à la page 145)**

Copiez l'objet (avec remplacement) (1, 4)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Rubrique	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE
Information de communication	MODIFICATION MQZAO_DE

Créer un *objet* (sans remplacement) (3)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(2)**
Rubrique	MQZAO_CREATE **(2)**
Processus	MQZAO_CREATE **(2)**
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_CREATE **(2)**
Informations d'authentification	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**
Canal de connexion client	MQZAO_CREATE **(2)**
Programme d'écoute	MQZAO_CREATE **(2)**
Service	MQZAO_CREATE **(2)**
Information de communication	MQZAO_CREATE **(2)**

Créer *objet* (avec remplacement) (3, 4)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Rubrique	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE
Information de communication	MODIFICATION MQZAO_DE

Supprimer *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DELETE
Rubrique	MQZAO_DELETE
Processus	MQZAO_DELETE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_DELETE
Informations d'authentification	MQZAO_DELETE
Canal	MQZAO_DELETE

Objet	Autorisation requise
Canal de connexion client	MQZAO_DELETE
Programme d'écoute	MQZAO_DELETE
Service	MQZAO_DELETE
Information de communication	MQZAO_DELETE

Interroger *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DISPLAY
Rubrique	MQZAO_DISPLAY
Processus	MQZAO_DISPLAY
Gestionnaire de files d'attente	MQZAO_DISPLAY
Liste de noms	MQZAO_DISPLAY
Informations d'authentification	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de connexion client	MQZAO_DISPLAY
Programme d'écoute	MQZAO_DISPLAY
Service	MQZAO_DISPLAY
Information de communication	MQZAO_DISPLAY

Consulter les noms d' *objet*

Objet	Autorisation requise
File d'attente	Aucun contrôle
Topic	Aucun contrôle
Processus	Aucun contrôle
Gestionnaire de files d'attente	Aucun contrôle
Liste de noms	Aucun contrôle
Informations d'authentification	Aucun contrôle
Canal	Aucun contrôle
Canal de connexion client	Aucun contrôle
Programme d'écoute	Aucun contrôle
Service	Aucun contrôle
Information de communication	Aucun contrôle

Démarrez *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable

Objet	Autorisation requise
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ
Information de communication	Non applicable

Arrêter *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ
Information de communication	Non applicable

Commandes relatives aux canaux

Commande	Objet	Autorisation requise
Ping Channel	Canal	CONTROLE MQZ
Reset Channel	Canal	MQZAO_CONTRÔLE_ÉTENDU
Resolve Channel	Canal	MQZAO_CONTRÔLE_ÉTENDU

Commandes d'abonnement

Commande	Objet	Autorisation requise
Modifier un abonnement	Topic	CONTROLE MQZ
Créer un abonnement	Topic	CONTROLE MQZ
Supprimer l'abonnement	Topic	CONTROLE MQZ
Consulter un abonnement	Topic	MQZAO_DISPLAY

Commandes de sécurité

Commande	Objet	Autorisation requise
Définition de l'enregistrement de droits d'accès	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Supprimer l'enregistrement de droits d'accès	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Consulter des enregistrements de droits	Gestionnaire de files d'attente	MQZAO_DISPLAY
Consulter un service de droits d'accès	Gestionnaire de files d'attente	MQZAO_DISPLAY
Interroger les droits d'accès de l'entité	Gestionnaire de files d'attente	MQZAO_DISPLAY
Définir l'enregistrement d'authentification de canal	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Consulter les enregistrements d'authentification de canal	Gestionnaire de files d'attente	MQZAO_DISPLAY
Régénérer la sécurité	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE

Affichages de statut

Commande	Objet	Autorisation requise
Inquire Channel Status	Gestionnaire de files d'attente	MQZAO_DISPLAY Notez que les droits `+inq` (ou de manière équivalente MQZAO_INQUIRE) sont requis sur la file d'attente de transmission si le type de canal est CLUSSDR.
Interroger le statut du programme d'écoute de canal	Gestionnaire de files d'attente	MQZAO_DISPLAY
Interroger le statut de publication / d'abonnement	Gestionnaire de files d'attente	MQZAO_DISPLAY
Interroger le statut de l'abonnement	Gestionnaire de files d'attente	MQZAO_DISPLAY
Consulter le statut d'un service	Gestionnaire de files d'attente	MQZAO_DISPLAY
Consulter le statut d'une rubrique	Gestionnaire de files d'attente	MQZAO_DISPLAY

Commandes relatives aux clusters

Commande	Objet	Autorisation requise
Inquire Cluster Queue Manager	Gestionnaire de files d'attente	MQZAO_DISPLAY
Refresh Cluster	Appartenance au groupe'mqm'requise	Appartenance au groupe'mqm'requise
Reset Cluster	Appartenance au groupe'mqm'requise	Appartenance au groupe'mqm'requise

Commande	Objet	Autorisation requise
Suspend Queue Manager Cluster	Appartenance au groupe'mqm'requise	Appartenance au groupe'mqm'requise
Resume Queue Manager Cluster	Appartenance au groupe'mqm'requise	Appartenance au groupe'mqm'requise

Autres commandes d'administration

Commande	Objet	Autorisation requise
Ping Queue Manager	Gestionnaire de files d'attente	MQZAO_DISPLAY
Refresh Queue Manager	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Reset Queue Manager	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Reset Queue Statistics	File d'attente	MQZAO_DISPLAY et MQZAO_CHANGE
Consulter une connexion	Gestionnaire de files d'attente	MQZAO_DISPLAY
Arrêter une connexion	Gestionnaire de files d'attente	MODIFICATION MQZAO_DE

Remarque :

1. Pour les commandes de copie, le droit MQZAO_DISPLAY est également requis pour l'objet From.
2. Les droits MQZAO_CREATE ne sont pas spécifiques à un objet ou à un type d'objet particulier. Le droit de création est accordé pour tous les objets d'un gestionnaire de files d'attente spécifié, en spécifiant un type d'objet QMGR dans la commande `setmqaut` .
3. Pour les commandes de création, les droits MQZAO_DISPLAY sont également requis pour le système SYSTEM.DEFAULT.* .
4. Ceci s'applique si l'objet à remplacer existe déjà. Si ce n'est pas le cas, la vérification est la même que pour la copie ou la création sans remplacement.

AIX Création et gestion de groupes sur AIX

Sous AIX, si vous n'utilisez pas NIS ou NIS +, utilisez SMITTY pour gérer les groupes.

Pourquoi et quand exécuter cette tâche

Sous AIX, vous pouvez utiliser SMITTY pour créer un groupe, ajouter un utilisateur à un groupe, afficher la liste des utilisateurs du groupe et supprimer un utilisateur d'un groupe.

Procédure

1. Dans SMITTY, sélectionnez **Sécurité et utilisateurs** et appuyez sur Entrée.
2. Sélectionnez **Groupes** et appuyez sur Entrée.
3. Pour créer un groupe, procédez comme suit:
   a) Sélectionnez **Ajouter un groupe** et appuyez sur Entrée.
   b) Entrez le nom du groupe et les noms des utilisateurs que vous souhaitez ajouter au groupe, séparés par des virgules.
   c) Appuyez sur Entrée pour créer le groupe.
4. Pour ajouter un utilisateur à un groupe, procédez comme suit:
   a) Sélectionnez **Modifier / Afficher les caractéristiques des groupes** et appuyez sur Entrée.
   b) Entrez le nom du groupe pour afficher la liste des membres du groupe.

c) Ajoutez les noms des utilisateurs que vous souhaitez ajouter au groupe, séparés par des virgules.

d) Appuyez sur Entrée pour ajouter les noms au groupe.

5. Pour afficher les membres d'un groupe, procédez comme suit:
   a) Sélectionnez **Modifier / Afficher les caractéristiques des groupes** et appuyez sur Entrée.
   b) Entrez le nom du groupe pour afficher la liste des membres du groupe.
6. Pour supprimer un utilisateur d'un groupe, procédez comme suit:
   a) Sélectionnez **Modifier / Afficher les caractéristiques des groupes** et appuyez sur Entrée.
   b) Entrez le nom du groupe pour afficher la liste des membres du groupe.
   c) Supprimez le nom de l'utilisateur que vous souhaitez supprimer du groupe.
   d) Appuyez sur Entrée pour supprimer le nom du groupe.

Linux Création et gestion de groupes sur Linux

Sous Linux, si vous n'utilisez pas NIS ou NIS +, utilisez le fichier `/etc/group` pour gérer les groupes.

Pourquoi et quand exécuter cette tâche

Sous Linux, les informations de groupe sont conservées dans le fichier `/etc/group` . Vous pouvez utiliser des commandes pour créer un groupe, ajouter un utilisateur à un groupe, afficher la liste des utilisateurs du groupe et supprimer un utilisateur d'un groupe.

Procédure

1. Pour créer un groupe, utilisez la commande **groupadd** .
   Entrez la commande suivante :

   ```
   groupadd -g group-ID group-name
   ```

   où *group-ID* est l'identificateur numérique du groupe et *group-name* est le nom du groupe.
2. Pour ajouter un membre à un groupe supplémentaire, utilisez la commande **usermod** pour répertorier les groupes supplémentaires dont l'utilisateur est actuellement membre, ainsi que les groupes supplémentaires dont l'utilisateur doit devenir membre.
   Par exemple, si l'utilisateur est déjà membre du groupe `groupa`et qu'il doit devenir membre de `groupb`, utilisez la commande suivante:

   ```
   usermod -G groupa,groupb user-name
   ```

   où *user-name* est le nom d'utilisateur.
3. Pour afficher les membres d'un groupe, utilisez la commande **getent** .
   Entrez la commande suivante :

   ```
   getent group group-name
   ```

   où *group-name* est le nom du groupe.
4. Pour supprimer un membre d'un groupe supplémentaire, utilisez la commande **usermod** pour répertorier les groupes supplémentaires dont vous souhaitez que l'utilisateur reste membre.
   Par exemple, si le groupe principal de l'utilisateur est `users` et que l'utilisateur est également membre des groupes `mqm`, `groupa` et `groupb`, pour supprimer l'utilisateur du groupe `mqm` , utilisez la commande suivante:

   ```
   usermod -G groupa,groupb user-name
   ```

   où *user-name* est le nom d'utilisateur.

Solaris Création et gestion de groupes sur Solaris

Sous Solaris, si vous n'utilisez pas NIS ou NIS +, utilisez le fichier `/etc/group` pour gérer les groupes.

Pourquoi et quand exécuter cette tâche

Sous Solaris, les informations de groupe sont conservées dans le fichier `/etc/group` . Vous pouvez utiliser des commandes pour créer un groupe, ajouter un utilisateur à un groupe, afficher la liste des utilisateurs du groupe et supprimer un utilisateur d'un groupe.

Procédure

1. Pour créer un groupe, utilisez la commande **groupadd** .
   Entrez la commande suivante :
   ```
   groupadd -g group-ID group-name
   ```
   où *group-ID* est l'identificateur numérique du groupe et *group-name* est le nom du groupe.
2. Pour ajouter un membre à un groupe supplémentaire, utilisez la commande **usermod** pour répertorier les groupes supplémentaires dont l'utilisateur est actuellement membre, ainsi que les groupes supplémentaires dont l'utilisateur doit devenir membre.
   Par exemple, si l'utilisateur est déjà membre du groupe `groupa`et qu'il doit devenir membre de `groupb`, utilisez la commande suivante:
   ```
   usermod -G groupa,groupb user-name
   ```
   où *user-name* est le nom d'utilisateur.
3. Pour savoir qui est membre d'un groupe, examinez l'entrée de ce groupe dans le fichier `/etc/group` .
4. Pour supprimer un membre d'un groupe supplémentaire, utilisez la commande **usermod** pour répertorier les groupes supplémentaires dont vous souhaitez que l'utilisateur reste membre.
   Par exemple, si le groupe principal de l'utilisateur est `users` et que l'utilisateur est également membre des groupes `mqm`, `groupa` et `groupb`, pour supprimer l'utilisateur du groupe `mqm` , utilisez la commande suivante:
   ```
   usermod -G groupa,groupb user-name
   ```
   où *user-name* est le nom d'utilisateur.

Windows Création et gestion de groupes sur Windows

Sous Windows, vous utilisez la fonction Gestion de l'ordinateur pour administrer des groupes sur un poste de travail ou une machine serveur membre.

Pourquoi et quand exécuter cette tâche

Pour les contrôleurs de domaine, les utilisateurs et les groupes sont administrés via Active Directory. Pour plus de détails sur l'utilisation d' Active Directory , reportez-vous aux instructions appropriées du système d'exploitation.

Les modifications apportées à l'appartenance à un groupe d'un principal ne sont pas reconnues tant que le gestionnaire de files d'attente n'est pas redémarré ou que vous n'émettez pas la commande MQSC **REFRESH SECURITY** (ou l'équivalent PCF).

Utilisez le panneau Gestion de l'ordinateur Windows pour gérer les utilisateurs et les groupes. Toute modification apportée à l'utilisateur actuellement connecté risque de ne pas être effective tant que l'utilisateur ne se reconnecte pas.

Windows *Création d'un groupe sous Windows*

Créez un groupe à l'aide du panneau de commande.

Procédure

1. Ouvrir le panneau de commande
2. Cliquez deux fois sur **Outils d'administration**.
   Le panneau Outils d'administration s'ouvre.
3. Cliquez deux fois sur **Gestion de l'ordinateur**.
   Le panneau Gestion de l'ordinateur s'ouvre.
4. Développez **Utilisateurs et groupes locaux**.
5. Cliquez avec le bouton droit de la souris sur **Groupes**et sélectionnez **Nouveau groupe ...**.
   Le panneau Nouveau groupe s'affiche.
6. Entrez un nom approprié dans la zone Nom du groupe, puis cliquez sur **Créer**.
7. Cliquez sur **Fermer**.

Windows *Ajout d'un utilisateur à un groupe sous Windows*

Ajoutez un utilisateur à un groupe à l'aide du panneau de commande.

Procédure

1. Ouvrir le panneau de commande
2. Cliquez deux fois sur **Outils d'administration**.
   Le panneau Outils d'administration s'ouvre.
3. Cliquez deux fois sur **Gestion de l'ordinateur**.
   Le panneau Gestion de l'ordinateur s'ouvre.
4. Dans le panneau Gestion de l'ordinateur, développez **Utilisateurs et groupes locaux**.
5. Sélectionnez **Utilisateurs**
6. Cliquez deux fois sur l'utilisateur que vous souhaitez ajouter à un groupe.
   Le panneau des propriétés utilisateur s'affiche.
7. Sélectionnez l'onglet **Membre de** .
8. Sélectionnez le groupe auquel vous souhaitez ajouter l'utilisateur. Si le groupe de votre choix n'est pas visible:
   a) Cliquez sur **Ajouter...**.
      Le panneau Sélectionner des groupes s'affiche.
   b) Cliquez sur **Emplacements ...**.
      Le panneau Emplacements s'affiche.
   c) Sélectionnez l'emplacement du groupe auquel vous souhaitez ajouter l'utilisateur dans la liste et cliquez sur **OK**.
   d) Entrez le nom du groupe dans la zone fournie.

      Vous pouvez également cliquer sur **Avancé ...** puis **Rechercher maintenant** pour répertorier les groupes disponibles dans l'emplacement actuellement sélectionné. A partir d'ici, sélectionnez le groupe auquel vous souhaitez ajouter l'utilisateur et cliquez sur **OK**.
   e) Cliquez sur **OK**.
      Le panneau des propriétés utilisateur s'affiche avec le groupe que vous avez ajouté.
   f) Sélectionnez le groupe.
9. Cliquez sur **OK**.
   Le panneau Gestion de l'ordinateur s'affiche.

Windows *Affichage des personnes faisant partie d'un groupe sur Windows*

Affichez les membres d'un groupe à l'aide du panneau de commande.

Procédure

1. Ouvrir le panneau de commande
2. Cliquez deux fois sur **Outils d'administration**.
   Le panneau Outils d'administration s'ouvre.
3. Cliquez deux fois sur **Gestion de l'ordinateur**.
   Le panneau Gestion de l'ordinateur s'ouvre.
4. Dans le panneau Gestion de l'ordinateur, développez **Utilisateurs et groupes locaux**.
5. Sélectionnez **Groupes**.
6. Cliquez deux fois sur un groupe. Le panneau des propriétés de groupe s'affiche.
   Le panneau des propriétés de groupe s'affiche.

Résultats

Les membres du groupe s'affichent.

Windows *Suppression d'un utilisateur d'un groupe sous Windows*

Supprimez un utilisateur d'un groupe à l'aide du panneau de commande.

Procédure

1. Ouvrir le panneau de commande
2. Cliquez deux fois sur **Outils d'administration**.
   Le panneau Outils d'administration s'ouvre.
3. Cliquez deux fois sur **Gestion de l'ordinateur**.
   Le panneau Gestion de l'ordinateur s'ouvre.
4. Dans le panneau Gestion de l'ordinateur, développez **Utilisateurs et groupes locaux**.
5. Sélectionnez **Utilisateurs**.
6. Cliquez deux fois sur l'utilisateur que vous souhaitez ajouter à un groupe.
   Le panneau des propriétés utilisateur s'affiche.
7. Sélectionnez l'onglet **Membre de** .
8. Sélectionnez le groupe dont vous souhaitez supprimer l'utilisateur, puis cliquez sur **Supprimer**.
9. Cliquez sur **OK**.
   Le panneau Gestion de l'ordinateur s'affiche.

Résultats

Vous venez de supprimer l'utilisateur du groupe.

Windows Remarques spéciales relatives à la sécurité sous Windows

Certaines fonctions de sécurité se comportent différemment sur les différentes versions de Windows.

La sécurité IBM MQ s'appuie sur les appels à l'API du système d'exploitation pour obtenir des informations sur les autorisations utilisateur et les appartenances à des groupes. Certaines fonctions ne se comportent pas de manière identique sur les systèmes Windows . Cette collection de rubriques comprend des descriptions de la manière dont ces différences peuvent affecter la sécurité IBM MQ lorsque vous exécutez IBM MQ dans un environnement Windows .

Windows *Comptes utilisateur locaux et de domaine pour le service IBM MQ Windows*

Lorsqu'IBM MQ s'exécute, il doit vérifier que seuls les utilisateurs autorisés peuvent accéder aux gestionnaires de files d'attente ou aux files d'attente. Cela nécessite un compte utilisateur spécial que IBM MQ peut utiliser pour demander des informations sur tout utilisateur qui tente d'accéder à ce type d'accès.

- «Configuration de comptes utilisateur spéciaux avec Prepare IBM MQ Wizard», à la page 150
- «Utilisation de IBM MQ avec Active Directory», à la page 150
- «Droits utilisateur requis pour un service IBM MQ Windows», à la page 151

Configuration de comptes utilisateur spéciaux avec Prepare IBM MQ Wizard

Prepare IBM MQ Wizard crée un compte utilisateur spécial afin que le service Windows puisse être partagé par les processus qui doivent l'utiliser (voir Configuration d' IBM MQ avec l'assistant de préparation d' IBM MQ).

Un service Windows est partagé entre les processus client pour une installation IBM MQ . Un service est créé pour chaque installation. Chaque service est nommé `MQ_`*`InstallationName`* et possède le nom d'affichage `IBM MQ(`*`InstallationName`*`)`.

Etant donné que chaque service doit être partagé entre des sessions de connexion non interactives et interactives, vous devez le lancer sous un compte utilisateur spécial. Vous pouvez utiliser un compte utilisateur spécial pour tous les services ou créer des comptes utilisateur spéciaux différents. Chaque compte utilisateur spécial doit avoir le droit de `se connecter en tant que service`. Pour plus d'informations, voir Tableau 14, à la page 151. Si l'ID utilisateur ne dispose pas des droits pour exécuter le service, ce dernier ne démarre pas et renvoie une erreur dans le journal des événements du système Windows. En règle générale, vous avez exécuté le Prepare IBM MQ Wizardet configuré l'ID utilisateur correctement. Toutefois, si vous avez configuré l'ID utilisateur manuellement, il se peut que vous ayez un problème à résoudre.

Lorsque vous installez IBM MQ et que vous exécutez Prepare IBM MQ Wizard pour la première fois, il crée un compte utilisateur local pour le service appelé MUSR_MQADMIN avec les paramètres et les droits requis, y compris `Connexion en tant que service`.

Pour les installations suivantes, Prepare IBM MQ Wizard crée un compte utilisateur nommé MUSR_MQADMIN*x*, où *x* est le prochain nombre disponible représentant un ID utilisateur qui n'existe pas. Le mot de passe de MUSR_MQADMIN*x* est généré de manière aléatoire lorsque le compte est créé et utilisé pour configurer l'environnement de connexion pour le service. Le mot de passe généré n'expire pas.

Ce compte IBM MQ n'est affecté par aucune règle de compte configurée sur le système pour exiger que les mots de passe de compte soient modifiés après une certaine période.

Le mot de passe n'est pas connu en dehors de ce traitement à utilisation unique et est stocké par le système d'exploitation Windows dans une partie sécurisée du registre.

Utilisation de IBM MQ avec Active Directory

Dans certaines configurations réseau, où les comptes utilisateur sont définis sur les contrôleurs de domaine qui utilisent le service d'annuaire Active Directory , le compte utilisateur local sous lequel IBM MQ s'exécute peut ne pas disposer des droits requis pour interroger l'appartenance à un groupe d'autres comptes utilisateur de domaine. Lorsque vous installez IBM MQ, Prepare IBM MQ Wizard identifie si tel est le cas en effectuant des tests et en vous posant des questions sur la configuration réseau.

Si le compte utilisateur local sous lequel IBM MQ s'exécute ne dispose pas des droits requis, le Prepare IBM MQ Wizard vous invite à indiquer les détails du compte d'un compte utilisateur de domaine avec des droits utilisateur particuliers. Pour plus d'informations sur la création et la configuration d'un compte de domaine Windows , voir Création et configuration de comptes de domaine Windows pour IBM MQ. Pour connaître les droits utilisateur requis par le compte utilisateur de domaine, voir Tableau 14, à la page 151.

Une fois que vous avez entré des détails de compte valides pour le compte utilisateur de domaine dans le Prepare IBM MQ Wizard, l'assistant configure un service IBM MQ Windows à exécuter sous le nouveau compte. Les détails du compte sont conservés dans la partie sécurisée du registre et ne peuvent pas être lus par les utilisateurs.

Lorsque le service est en cours d'exécution, un service IBM MQ Windows est lancé et reste en cours d'exécution tant que le service est en cours d'exécution. Un administrateur IBM MQ qui se connecte au serveur après le lancement du service Windows peut utiliser IBM MQ Explorer pour administrer les

gestionnaires de files d'attente sur le serveur. Le IBM MQ Explorer est ainsi connecté au processus de service Windows existant. Ces deux actions nécessitent des niveaux d'autorisation différents pour pouvoir fonctionner:

- Le processus de lancement requiert un droit de lancement.
- L'administrateur IBM MQ requiert des droits d'accès.

Droits utilisateur requis pour un service IBM MQ Windows

Le tableau suivant répertorie les droits utilisateur requis pour les comptes utilisateur locaux et de domaine sous lesquels s'exécute le service Windows pour une installation IBM MQ .

Tableau 14. Droits utilisateur requis pour un service Windows IBM MQ

Droit	**Description**
Se connecter en tant que travail par lots	Active un service IBM MQ Windows à exécuter sous ce compte utilisateur.
Ouvrir une session en tant que service	Permet aux utilisateurs de définir le service IBM MQ Windows pour se connecter à l'aide du compte configuré.
Arrêter le système	Permet au service IBM MQ Windows de redémarrer le serveur s'il est configuré pour le faire en cas d'échec de la reprise d'un service.
Augmenter les quotas	Obligatoire pour l'appel `CreateProcessAsUser` du système d'exploitation.
Agir comme partie intégrante du système d'exploitation	Obligatoire pour l'appel `LogonUser` du système d'exploitation.
Ignorer le contrôle transversal	Obligatoire pour l'appel `LogonUser` du système d'exploitation.
Remplacer un jeton de processus	Obligatoire pour l'appel `LogonUser` du système d'exploitation.

Remarque : Les droits des programmes de débogage peuvent être nécessaires dans les environnements exécutant des applications ASP et IIS.

Votre compte utilisateur de domaine doit avoir ces droits d'utilisateur Windows définis comme droits d'utilisateur effectifs, comme indiqué dans l'application Stratégie de sécurité locale. Si ce n'est pas le cas, définissez-les à l'aide de l'application Stratégie de sécurité locale en local sur le serveur ou à l'aide du domaine d'application de sécurité du domaine.

Windows *Droits de sécurité du serveur Windows*

L'installation de IBM MQ se comporte différemment sur Windows Server, selon qu'un utilisateur local ou un utilisateur de domaine effectue l'installation.

Si un utilisateur *local* installe IBM MQ, Prepare IBM MQ Wizard détecte que l'utilisateur local créé pour le service IBM MQ Windows peut extraire les informations d'appartenance à un groupe de l'utilisateur installant. Le Prepare IBM MQ Wizard pose des questions à l'utilisateur sur la configuration réseau afin de déterminer si d'autres comptes utilisateur sont définis sur les contrôleurs de domaine s'exécutant sous Windows 2000 ou version ultérieure. Si tel est le cas, le service IBM MQ Windows doit s'exécuter sous un compte utilisateur de domaine avec des paramètres et des droits spécifiques. Prepare IBM MQ Wizard invite l'utilisateur à entrer les détails de son compte, comme décrit dans Configuration d' IBM MQ à l'aide de l'assistant de préparation d' IBM MQ.

Si un utilisateur *domain* installe IBM MQ, Prepare IBM MQ Wizard détecte que l'utilisateur local créé pour le service IBM MQ Windows ne peut pas extraire les informations d'appartenance à un groupe de

l'utilisateur installant. Dans ce cas, Prepare IBM MQ Wizard invite toujours l'utilisateur à indiquer les détails du compte utilisateur de domaine à utiliser par le service IBM MQ Windows .

Lorsque le service IBM MQ Windows doit utiliser un compte utilisateur de domaine, IBM MQ ne peut pas fonctionner correctement tant qu'il n'a pas été configuré à l'aide de Prepare IBM MQ Wizard. Prepare IBM MQ Wizard ne permet pas à l'utilisateur de continuer avec d'autres tâches tant que le service Windows n'a pas été configuré avec un compte approprié.

Pour plus d'informations, voir Création et configuration de comptes de domaine pour IBM MQ.

Windows *Modification du nom d'utilisateur associé au service IBM MQ*

Vous pouvez modifier le nom d'utilisateur associé au service IBM MQ en créant un nouveau compte et en entrant ses détails à l'aide du Prepare IBM MQ Wizard.

Pourquoi et quand exécuter cette tâche

Lorsque vous installez IBM MQ et exécutez Prepare IBM MQ Wizard pour la première fois, il crée un compte utilisateur local pour le service appelé MUSR_MQADMIN. Pour les installations suivantes, Prepare IBM MQ Wizard crée un compte utilisateur nommé MUSR_MQADMIN*x*, où *x* est le prochain nombre disponible représentant un ID utilisateur qui n'existe pas.

Vous devrez peut-être changer le nom d'utilisateur associé au service IBM MQ de MUSR_MQADMIN ou MUSR_MQADMIN*x* à autre chose. Par exemple, vous pouvez être amené à effectuer cette opération si votre gestionnaire de files d'attente est associé à Db2, qui n'accepte pas les noms d'utilisateur de plus de 8 caractères.

Procédure

1. Créer un nouveau compte utilisateur (par exemple, **NEW_NAME**)
2. Utilisez le Prepare IBM MQ Wizard pour entrer les détails du nouveau compte utilisateur.

Tâches associées

Configuration d'IBM MQ avec l'assistant de préparation de IBM MQ

Windows *Modification du mot de passe du compte utilisateur local du service IBM MQ Windows*

Vous pouvez modifier le mot de passe du compte utilisateur local du service IBM MQ Windows à l'aide du panneau Gestion de l'ordinateur.

Pourquoi et quand exécuter cette tâche

Pour modifier le mot de passe du compte utilisateur local du service IBM MQ Windows , procédez comme suit:

Procédure

1. Identifiez l'utilisateur sous lequel le service s'exécute.
2. Arrêtez le service IBM MQ à partir du panneau Gestion de l'ordinateur.
3. Modifiez le mot de passe requis de la même manière que vous modifiez le mot de passe d'une personne.
4. Accédez aux propriétés du service IBM MQ à partir du panneau Gestion de l'ordinateur.
5. Sélectionnez la page **Connexion** .
6. Vérifiez que le nom de compte spécifié correspond à l'utilisateur pour lequel le mot de passe a été modifié.
7. Entrez le mot de passe dans les zones **Mot de passe** et **Confirmer le mot de passe** , puis cliquez sur **OK**.

Windows *Modification du mot de passe d'un service IBM MQ Windows pour une installation exécutée sous un compte utilisateur de domaine*

Comme alternative à l'utilisation du Prepare IBM MQ Wizard pour entrer les détails du compte utilisateur de domaine, vous pouvez utiliser le panneau Gestion de l'ordinateur pour modifier les détails de la **connexion** pour le service IBM MQ spécifique à l'installation.

Pourquoi et quand exécuter cette tâche

Si le service IBM MQ Windows d'une installation s'exécute sous un compte utilisateur de domaine, vous pouvez modifier le mot de passe du compte comme suit:

Procédure

1. Modifiez le mot de passe du compte de domaine sur le contrôleur de domaine. Vous devrez peut-être demander à votre administrateur de domaine de le faire pour vous.
2. Procédez comme suit pour modifier la page **Connexion** du service IBM MQ .
   a) Identifiez l'utilisateur sous lequel le service s'exécute.
   b) Arrêtez le service IBM MQ à partir du panneau Gestion de l'ordinateur.
   c) Modifiez le mot de passe requis de la même manière que vous modifiez le mot de passe d'une personne.
   d) Accédez aux propriétés du service IBM MQ à partir du panneau Gestion de l'ordinateur.
   e) Sélectionnez la page **Connexion** .
   f) Vérifiez que le nom de compte spécifié correspond à l'utilisateur pour lequel le mot de passe a été modifié.
   g) Entrez le mot de passe dans les zones **Mot de passe** et **Confirmer le mot de passe** , puis cliquez sur **OK**.

   Le compte utilisateur sous lequel le service IBM MQ Windows s'exécute exécute exécute toutes les commandes MQSC qui sont émises par les applications de l'interface utilisateur ou qui sont exécutées automatiquement lors du démarrage, de l'arrêt ou de la reprise du service du système. Ce compte utilisateur doit donc disposer des droits d'administration IBM MQ . Par défaut, il est ajouté au groupe mqm local sur le serveur. Si cette appartenance est supprimée, le service IBM MQ Windows ne fonctionne pas. Pour plus d'informations sur les droits utilisateur, voir «Droits utilisateur requis pour un service IBM MQ Windows», à la page 151.

   Si un problème de sécurité survient avec le compte utilisateur sous lequel le service IBM MQ Windows s'exécute, des messages d'erreur et des descriptions apparaissent dans le journal des événements système.

Tâches associées

Configuration d'IBM MQ avec l'assistant de préparation de IBM MQ

Windows ***Remarques à prendre en compte lors de la promotion de serveurs Windows vers des contrôleurs de domaine***

Lors de la promotion d'un serveur Windows sur un contrôleur de domaine, vous devez déterminer si le paramètre de sécurité relatif aux droits d'accès des utilisateurs et des groupes est approprié. Lorsque vous modifiez l'état d'une machine Windows entre le serveur et le contrôleur de domaine, vous devez tenir compte du fait que cela peut affecter le fonctionnement de IBM MQ car IBM MQ utilise un groupe mqm défini en local.

Paramètres de sécurité relatifs aux droits d'utilisateur de domaine et de groupe

IBM MQ s'appuie sur les informations d'appartenance à un groupe pour implémenter sa stratégie de sécurité, ce qui signifie qu'il est important que l'ID utilisateur qui exécute des opérations IBM MQ puisse déterminer les appartenances à des groupes d'autres utilisateurs.

Lorsque vous promouvez un serveur Windows sur un contrôleur de domaine, une option s'affiche pour le paramètre de sécurité relatif aux droits d'accès des utilisateurs et des groupes. Cette option contrôle si des utilisateurs arbitraires peuvent extraire des appartenances à des groupes à partir du répertoire actif. Si un contrôleur de domaine est configuré de sorte que les comptes locaux soient autorisés à interroger l'appartenance à un groupe des comptes utilisateur de domaine, l'ID utilisateur par défaut créé par IBM MQ lors du processus d'installation peut obtenir des appartenances à des groupes pour d'autres utilisateurs, selon les besoins. Toutefois, si un contrôleur de domaine est configuré de sorte que les comptes locaux ne soient pas autorisés à interroger l'appartenance à un groupe des comptes utilisateur de domaine, cela empêche IBM MQ de vérifier que les utilisateurs définis sur le domaine sont autorisés à accéder aux gestionnaires de files d'attente ou aux files d'attente et que l'accès échoue. Si vous utilisez Windows sur un contrôleur de domaine qui a été configuré de cette manière, un compte utilisateur de domaine spécial avec les droits requis doit être utilisé.

Dans ce cas, vous devez connaître:

- Comment se comportent les droits de sécurité pour votre version de Windows .
- Comment autoriser les membres du groupe mqm de domaine à lire l'appartenance à un groupe.
- Comment configurer un service IBM MQ Windows pour qu'il s'exécute sous un utilisateur de domaine.

Pour plus d'informations, voir Configuration de comptes utilisateur pour IBM MQ.

Accès IBM MQ au groupe mqm local

Lorsque des serveurs Windows sont promus ou rétrogradés à partir de contrôleurs de domaine, IBM MQ perd l'accès au groupe mqm local.

Lorsqu'un serveur est promu en tant que contrôleur de domaine, la portée passe de locale à locale. Lorsque la machine est rétrogradée sur le serveur, tous les groupes locaux de domaine sont supprimés. Cela signifie que le passage d'une machine d'un serveur à un contrôleur de domaine et le retour au serveur perdent l'accès à un groupe mqm local. Le symptôme est une erreur indiquant l'absence d'un groupe mqm local, par exemple:

```
>crtmqm qm0
AMQ8066:Local mqm group not found.
```

Pour résoudre ce problème, recréez le groupe mqm local à l'aide des outils de gestion Windows standard. Etant donné que toutes les informations d'appartenance à un groupe sont perdues, vous devez rétablir les utilisateurs IBM MQ privilégiés dans le groupe mqm local nouvellement créé. Si la machine est un membre de domaine, vous devez également ajouter le groupe mqm de domaine au groupe mqm local pour accorder aux ID utilisateur IBM MQ de domaine privilégié le niveau de droits requis.

Windows *Restrictions sur les groupes imbriqués sous Windows*

L'utilisation de groupes imbriqués est soumise à des restrictions. Elles résultent en partie du niveau fonctionnel du domaine et en partie des restrictions IBM MQ .

Active Directory peut prendre en charge différents types de groupe dans un contexte de domaine en fonction du niveau fonctionnel du domaine. Par défaut, les domaines Windows 2003 se trouvent dans le répertoire " Niveau fonctionnel Windows 2000 mixte. (Windows Server 2008 et Windows Server 2012 suivent le modèle de domaine Windows 2003 .) Le niveau fonctionnel de domaine détermine les types de groupe pris en charge et le niveau d'imbrication autorisé lors de la configuration des ID utilisateur dans un environnement de domaine. Reportez-vous à la documentation Active Directory pour plus de détails sur la portée du groupe et les critères d'inclusion.

Outre les exigences relatives à Active Directory , d'autres restrictions s'appliquent aux ID utilisés par IBM MQ. Les API réseau utilisées par IBM MQ ne prennent pas en charge toutes les configurations prises en charge par le niveau fonctionnel de domaine. Par conséquent, IBM MQ ne peut pas interroger les appartenances aux groupes des ID de domaine présents dans un groupe local de domaine qui est ensuite imbriqué dans un groupe local. En outre, l'imbrication multiple de groupes globaux et universels n'est pas prise en charge. Toutefois, les groupes globaux ou universels immédiatement imbriqués sont pris en charge.

Windows *Autorisation des utilisateurs à utiliser IBM MQ à distance*

Si vous devez créer et démarrer des gestionnaires de files d'attente lorsque vous êtes connecté à IBM MQ à distance, vous devez disposer de l'accès utilisateur `Créer des objets globaux` .

Pourquoi et quand exécuter cette tâche

Remarque : Les administrateurs disposent de l'accès de `création d'objets globaux` par défaut. Par conséquent, si vous en êtes un, vous pourrez créer et démarrer des gestionnaires de files d'attente en étant connecté à distance sans modifier vos droits utilisateur.

Si vous vous connectez à une machine Windows à l'aide de Terminal Services ou d'une connexion Bureau à distance et que vous rencontrez des problèmes lors de la création, du démarrage ou de la suppression d'un gestionnaire de files d'attente, cela peut être dû au fait que vous ne disposez pas de l'accès utilisateur `Créer des objets globaux`.

L'accès utilisateur de `création d'objets globaux` limite les utilisateurs autorisés à créer des objets dans l'espace de nom global. Pour qu'une application crée un objet global, elle doit s'exécuter dans un espace de nom global ou l'utilisateur sous l'ID duquel s'exécute l'application doit disposer de l'accès utilisateur de `création d'objets globaux`.

Lorsque vous vous connectez à distance à une machine Windows à l'aide des services Terminal Services ou d'une connexion de bureau distante, les applications s'exécutent dans leur propre espace de nom local. Si vous tentez de créer ou de supprimer un gestionnaire de files d'attente à l'aide d'IBM MQ Explorer ou avec la commande **crtmqm** ou **dltmqm**, ou si vous tentez de démarrer un gestionnaire de files d'attente avec la commande **strmqm**, un incident d'autorisation survient. Un FDC IBM MQ avec l'ID sonde XY132002 est créé.

Le démarrage d'un gestionnaire de files d'attente via IBM MQ Explorer ou avec la commande **amqmdain qmgr start** fonctionne car ces commandes ne démarrent pas le gestionnaire directement. En effet, ces commandes envoient la demande de démarrage du gestionnaire de files d'attente à un processus distinct s'exécutant dans l'espace de nom global.

Si les différentes méthodes d'administration de IBM MQ ne fonctionnent pas lorsque vous utilisez les services de terminal, essayez de définir le droit utilisateur `Créer des objets globaux` .

Procédure

1. Ouvrez le panneau Outils d'administration:

   Windows Server 2008 et Windows Server 2012
   Accédez à ce panneau à l'aide du **Panneau de configuration** > **Système et maintenance** > **Outils d'administration**.

   Windows 8.1
   Accédez à ce panneau à l'aide de **Outils d'administration** > **Gestion de l'ordinateur**
2. Cliquez deux fois sur **Stratégie de sécurité locale**.
3. Développez **Stratégies locales**.
4. Cliquez sur **Affectation des droits utilisateur**.
5. Ajoutez le nouvel utilisateur ou le nouveau groupe à la règle `Créer des objets globaux` .

Windows *Programme d'exit de canal SSPI sous Windows*

IBM MQ for Windows fournit un programme d'exit de sécurité, qui peut être utilisé sur les canaux de message et MQI. L'exit est fourni en tant que code source et code objet et fournit une authentification unidirectionnelle et bidirectionnelle.

L'exit de sécurité utilise l'interface SSPI (Security Support Provider Interface), qui fournit les fonctions de sécurité intégrées des plateformes Windows .

L'exit de sécurité fournit les services d'identification et d'authentification suivants:

authentification unidirectionnelle
Cela utilise la prise en charge de l'authentification Windows NT LAN Manager (NTLM). NTLM permet aux serveurs d'authentifier leurs clients. Il ne permet pas à un client d'authentifier un serveur ou à un serveur d'en authentifier un autre. NTLM a été conçu pour un environnement réseau dans lequel les serveurs sont supposés être authentiques. NTLM est pris en charge sur toutes les plateformes Windows prises en charge par IBM WebSphere MQ 7.0.

Ce service est généralement utilisé sur un canal MQI pour permettre à un gestionnaire de files d'attente serveur d'authentifier une application IBM MQ MQI client . Une application client est identifiée par l'ID utilisateur associé au processus en cours d'exécution.

Pour effectuer l'authentification, l'exit de sécurité à l'extrémité client d'un canal acquiert un jeton d'authentification auprès de NTLM et envoie le jeton dans un message de sécurité à son partenaire à l'autre extrémité du canal. L'exit de sécurité partenaire transmet le jeton à NTLM, qui vérifie que le jeton est authentique. Si l'exit de sécurité partenaire n'est pas satisfait de l'authenticité du jeton, il demande à l'agent MCA de fermer le canal.

Authentification bidirectionnelle ou mutuelle
Cela utilise les services d'authentification Kerberos . Le protocole Kerberos ne suppose pas que les serveurs d'un environnement réseau sont authentiques. Les serveurs peuvent authentifier les clients et d'autres serveurs, et les clients peuvent authentifier les serveurs. Kerberos est pris en charge sur toutes les plateformes Windows prises en charge par IBM WebSphere MQ 7.0.

Ce service peut être utilisé sur les canaux de message et MQI. Sur un canal de transmission de messages, il fournit une authentification mutuelle des deux gestionnaires de files d'attente. Sur un canal MQI, il permet au gestionnaire de files d'attente du serveur et à l'application IBM MQ MQI client de s'authentifier mutuellement. Un gestionnaire de files d'attente est identifié par son nom préfixé par la chaîne `ibmMQSeries/`. Une application client est identifiée par l'ID utilisateur associé au processus en cours d'exécution.

Pour effectuer l'authentification mutuelle, l'exit de sécurité initiateur acquiert un jeton d'authentification auprès du serveur de sécurité Kerberos et envoie le jeton dans un message de sécurité à son partenaire. L'exit de sécurité partenaire transmet le jeton au serveur Kerberos , qui vérifie qu'il est authentique. Le serveur de sécurité Kerberos génère un second jeton que le partenaire envoie dans un message de sécurité à l'exit de sécurité initiateur. L'exit de sécurité initiateur demande ensuite au serveur Kerberos de vérifier que le deuxième jeton est authentique. Lors de cet échange, si l'un des exits de sécurité n'est pas satisfait de l'authenticité du jeton envoyé par l'autre, il demande à l'agent MCA de fermer le canal.

L'exit de sécurité est fourni au format source et au format objet. Vous pouvez utiliser le code source comme point de départ pour l'écriture de vos propres programmes d'exit de canal ou vous pouvez utiliser le module d'objet tel qu'il est fourni. Le module d'objet possède deux points d'entrée, l'un pour l'authentification unidirectionnelle à l'aide de la prise en charge de l'authentification NTLM et l'autre pour l'authentification bidirectionnelle à l'aide des services d'authentification Kerberos .

Pour plus d'informations sur le fonctionnement du programme d'exit de canal SSPI et pour savoir comment l'implémenter, voir Utilisation de l'exit de sécurité SSPI sur les systèmes Windows.

Windows *Application des fichiers de modèle de sécurité sous Windows*

L'application d'un modèle peut affecter les paramètres de sécurité appliqués aux fichiers et répertoires IBM MQ . Si vous utilisez le modèle hautement sécurisé, appliquez-le avant d'installer IBM MQ.

Windows prend en charge les fichiers de modèle de sécurité textuelle que vous pouvez utiliser pour appliquer des paramètres de sécurité uniformes à un ou plusieurs ordinateurs avec le composant logiciel enfichable MMC Configuration et analyse de la sécurité. En particulier, Windows fournit plusieurs modèles qui incluent une série de paramètres de sécurité dans le but de fournir des niveaux de sécurité spécifiques. Ces modèles incluent Compatible, Secure et Hautement Secure.

L'application de l'un de ces modèles peut affecter les paramètres de sécurité appliqués aux fichiers et répertoires IBM MQ . Si vous souhaitez utiliser le modèle hautement sécurisé, configurez votre machine avant d'installer IBM MQ.

Si vous appliquez le modèle hautement sécurisé à une machine sur laquelle IBM MQ est déjà installé, tous les droits que vous avez définis sur les fichiers et répertoires IBM MQ sont supprimés. Etant donné que ces droits sont supprimés, vous perdez *Administrator*, *mqm*et, le cas échéant, l'accès du groupe *Everyone* à partir des répertoires d'erreurs.

Windows *Configuration de droits d'accès supplémentaires pour les applications Windows se connectant à IBM MQ*

Le compte sous lequel les processus IBM MQ s'exécutent peut nécessiter une autorisation supplémentaire pour que l'accès à SYNCHRONISER aux processus d'application puisse être accordé.

Pourquoi et quand exécuter cette tâche

Vous pouvez rencontrer des problèmes si vous avez des applications Windows , par exemple des pages ASP, qui se connectent à IBM MQ et qui sont configurées pour s'exécuter à un niveau de sécurité supérieur à la normale.

IBM MQ requiert l'accès SYNCHRONISER aux processus d'application afin de coordonner certaines actions. Lorsqu'une application serveur tente pour la première fois de se connecter à un gestionnaire de files d'attente, IBM MQ modifie le processus pour accorder le droit SYNCHRONISER aux administrateurs IBM MQ . Toutefois, le compte sous lequel les processus IBM MQ s'exécutent peut nécessiter une autorisation supplémentaire pour que l'accès demandé puisse être accordé.

Pour configurer des droits supplémentaires sur l'ID utilisateur sous lequel les processus IBM MQ s'exécutent, procédez comme suit:

Procédure

1. Démarrez l'outil Stratégie de sécurité locale, cliquez sur **Paramètres de sécurité**->**Stratégies locales**->**Affectations de droits utilisateur**, puis sur **Déboguer les programmes**.
2. Cliquez deux fois sur **Déboguer les programmes**, puis ajoutez votre ID utilisateur IBM MQ à la liste

   Si le système se trouve dans un domaine Windows et que le paramètre de stratégie effectif n'est toujours pas défini, même si le paramètre de stratégie locale est défini, l'ID utilisateur doit être autorisé de la même manière au niveau du domaine, à l'aide de l'outil de stratégie de sécurité du domaine.

IBM i Configuration de la sécurité sous IBM i

La sécurité sur IBM i est implémentée à l'aide de IBM MQ Object Authority Manager (OAM) et de la sécurité au niveau de l'objet IBM i .

Considérations de sécurité à prendre en compte lors de la détermination des droits d'accès aux objets IBM MQ .

Vous devez prendre en compte les points suivants lors de la configuration des droits des utilisateurs de votre entreprise:

1. Accordez et révoquez les droits sur les commandes IBM MQ for IBM i à l'aide des commandes IBM i GRTOBJAUT et RVKOBJAUT .

   Dans la bibliothèque QMQM , certains objets noncommand (* cmd) sont définis pour disposer des droits ***PUBLIC** sur ***USE**. Ne modifiez pas les droits de ces objets et n'utilisez pas de liste d'autorisation pour fournir des droits. Toute autorité incorrecte peut compromettre la fonctionnalité IBM MQ .
2. Lors de l'installation de IBM MQ for IBM i, les profils utilisateur spéciaux suivants sont créés:

   QMQM
   Est utilisé principalement pour les fonctions internes du produit uniquement. Toutefois, il peut être utilisé pour exécuter des applications sécurisées à l'aide de MQCNO_FASTPATH_BINDINGS. Voir Connexion à un gestionnaire de files d'attente à l'aide de l'appel MQCONNX.

QMQMADM
Est utilisé comme profil de groupe pour les administrateurs de IBM MQ. Le profil de groupe donne accès aux commandes CL et aux ressources IBM MQ .

Lors de l'utilisation de SBMJOB pour soumettre des programmes qui appellent des commandes IBM MQ , USER ne doit pas être défini explicitement sur QMQMADM. A la place, définissez USER sur QMQM ou sur un autre profil utilisateur pour lequel QMQMADM est spécifié en tant que groupe.

3. Si vous envoyez des commandes de canal à des gestionnaires de files d'attente éloignées, vérifiez que votre profil utilisateur est membre du groupe QMQMADM sur le système cible. Pour obtenir la liste des commandes de canal PCF et MQSC, voir Commandes CLIBM MQ for IBM i.
4. L'ensemble de groupes associé à un utilisateur est mis en cache lorsque les autorisations de groupe sont calculées par la méthode d'accès aux objets (OAM).

   Les modifications apportées aux appartenances à un groupe d'utilisateurs après la mise en cache de l'ensemble de groupes ne sont pas reconnues tant que vous n'avez pas redémarré le gestionnaire de files d'attente ou exécuté RFRMQMAUT pour actualiser la sécurité.
5. Limitez le nombre d'utilisateurs autorisés à utiliser des commandes particulièrement sensibles. Ces commandes incluent:
   - Création d'un gestionnaire de files d'attente de messages (CRTMQM)
   - Supprimer le gestionnaire de files d'attente de messages (DLTMQM)
   - Démarrage du gestionnaire de files d'attente de messages (STRMQM)
   - Arrêt du gestionnaire de files d'attente de messages (ENDMQM)
   - Démarrer le serveur de commandes (STRMQMCSVR)
   - Arrêt du serveur de commandes (ENDMQMCSVR)
6. Les définitions de canal contiennent une spécification de programme d'exit de sécurité. La création et la modification de canaux nécessitent des considérations particulières. Les détails des exits de sécurité sont fournis dans «Présentation de l'exit de sécurité», à la page 108.
7. Les programmes d'exit de canal et de moniteur de déclenchement peuvent être remplacés. La sécurité de ces remplacements est de la responsabilité du programmeur.

IBM i Gestionnaire des droits d'accès aux objets sous IBM i

Le gestionnaire des droits d'accès aux objets (OAM) gère les autorisations des utilisateurs pour manipuler les objets IBM MQ , y compris les files d'attente et les définitions de processus. Il fournit également une interface de commande grâce à laquelle vous pouvez accorder ou révoquer des droits d'accès à un objet pour un groupe spécifique d'utilisateurs. La décision d'autoriser l'accès à une ressource est prise par la méthode d'accès aux objets (OAM) et le gestionnaire de files d'attente suit cette décision. Si la méthode d'accès aux objets (OAM) ne peut pas prendre de décision, le gestionnaire de files d'attente empêche l'accès à cette ressource.

Grâce à la méthode d'accès aux objets (OAM), vous pouvez contrôler:

- Accès aux objets IBM MQ via l'interface MQI. Lorsqu'un programme d'application tente d'accéder à un objet, la méthode d'accès aux objets (OAM) vérifie que le profil utilisateur à l'origine de la demande dispose de l'autorisation pour l'opération demandée.

  En particulier, cela signifie que les files d'attente et les messages des files d'attente peuvent être protégés contre les accès non autorisés.
- Droit d'utilisation des commandes PCF et MQSC.

Différents groupes d'utilisateurs peuvent disposer de droits d'accès différents sur le même objet. Par exemple, pour une file d'attente spécifique, un groupe peut effectuer à la fois des opérations d'insertion et d'extraction ; un autre groupe peut être autorisé uniquement à parcourir la file d'attente (MQGET avec l'option de navigation). De même, certains groupes peuvent avoir des droits d'extraction et d'insertion sur une file d'attente, mais ils ne sont pas autorisés à modifier ou à supprimer la file d'attente.

Commandes IBM MQ for IBM i et opérations sur les objets IBM MQ for IBM i

IBM i Droits IBM MQ sur IBM i

Pour accéder aux objets IBM MQ , vous devez disposer des droits permettant d'émettre la commande et d'accéder à l'objet référencé. Les administrateurs ont accès à toutes les ressources IBM MQ .

L'accès aux objets IBM MQ est contrôlé par les droits d'accès à:

1. Exécutez la commande IBM MQ
2. Accès aux objets IBM MQ référencés par la commande

Toutes les commandes CL IBM MQ for IBM i sont fournies avec un propriétaire de QMQM et le profil d'administration (QMQMADM) possède les droits *USE avec l'accès *PUBLIC défini sur *EXCLUDE.

Remarque : Le programme QSRDUPER est utilisé par le programme d'installation de logiciel sous licence IBM MQ for IBM i pour dupliquer des objets Commande (*CMD) dans QSYS. Dans IBM i V5R4 et les versions ultérieures, le programme QSRDUPER a été modifié de sorte que le comportement par défaut consiste à créer une commande proxy plutôt qu'un doublon de la commande d'origine. Une commande proxy redirige l'exécution de la commande vers une autre commande et possède un attribut PRX. Si une commande proxy portant le même nom que la commande en cours de copie existe dans la bibliothèque QSYS, les droits privés sur la commande proxy ne sont pas accordés à la commande dans la bibliothèque du produit. Les tentatives d'invite ou d'exécution de la commande proxy dans QSYS vérifient les droits de la commande cible dans la bibliothèque du produit. Toute modification des droits sur les objets *CMD doit donc être effectuée dans la bibliothèque de produit (QMQM) et celles de QSYS n'ont pas besoin d'être modifiées. Exemple :

```
GRTOBJAUT OBJ(QMQM/DSPMQMQ) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
```

Les modifications apportées à la structure des droits d'accès de certaines des commandes CL du produit permettent l'utilisation publique de ces commandes, si vous disposez des droits d'accès OAM requis sur les objets IBM MQ pour effectuer ces modifications.

Pour être un administrateur IBM MQ sous IBM i, vous devez être membre du *groupe QMQMADM*. Ce groupe possède des propriétés telles que les propriétés du groupe mqm sur les systèmes UNIX, Linux et Windows . En particulier, le groupe QMQMADM est créé lorsque vous installez IBM MQ for IBM iet les membres du groupe QMQMADM ont accès à toutes les ressources IBM MQ sur le système. Vous avez également accès à toutes les ressources IBM MQ si vous disposez des droits *ALLOBJ.

Les administrateurs peuvent utiliser des commandes CL pour administrer IBM MQ. L'une de ces commandes est GRTMQMAUT, qui permet d'accorder des droits à d'autres utilisateurs. Une autre commande, STRMQMMQSC, permet à un administrateur d'émettre des commandes MQSC vers un gestionnaire de files d'attente local.

Concepts associés

«Droit d'administration de IBM MQ sur IBM i», à la page 86

IBM i *Droits d'accès pour les objets IBM MQ sous IBM i*

Droits d'accès requis pour l'exécution des commandes CL IBM MQ .

IBM MQ for IBM i catégorise les commandes CL du produit en deux groupes:

Groupe 1

Les utilisateurs doivent appartenir au groupe d'utilisateurs QMQMADM ou disposer des droits *ALLOBJ pour pouvoir traiter ces commandes. Les utilisateurs disposant de l'un ou l'autre de ces droits peuvent traiter toutes les commandes de toutes les catégories sans avoir besoin de droits supplémentaires.

Remarque : Ces droits remplacent les droits OAM.

Ces commandes peuvent être regroupées comme suit:

- Commandes relatives au serveur de commandes
  - ENDMQMCSVR, arrêt du serveur de commandes IBM MQ

  - STRMQMCSVR, démarrage du serveur de commandes IBM MQ
- Commande du gestionnaire de files d'attente de messages non livrés
  - STRMQMDLQ, démarrage du gestionnaire de file d'attente des messages non livrés IBM MQ
- Commande du programme d'écoute
  - ENDMQMLSR, Arrêt du programme d'écoute IBM MQ
  - STRMQMLSR, démarrage d'un programme d'écoute non-objet
- Commandes relatives à la reprise sur incident lié au support
  - RCDMQMIMG, Enregistrement d'image d'objet IBM MQ
  - RCRMQMOBJ, recréation d'objet IBM MQ
  - WRKMQMTRN, Utilisation des transactions IBM MQ Q
- Commandes relatives au gestionnaire de files d'attente
  - CRTMQM, Création d'un gestionnaire de files d'attente de messages
  - DLTMQM, Suppression d'un gestionnaire de files d'attente de messages
  - ENDMQM, Arrêt du gestionnaire de files d'attente de messages
  - STRMQM, démarrage du gestionnaire de files d'attente de messages
- Commandes de sécurité
  - GRTMQMAUT, octroi de droits sur les objets IBM MQ
  - RVKMQMAUT, révocation des droits sur les objets IBM MQ
- Commande relative aux traces
  - TRCMQM, Traçage du travail IBM MQ
- Commandes de transaction
  - RSVMQMTRN, Résolution de la transaction IBM MQ
- Commandes moniteur de déclenchement
  - STRMQMTRM, démarrage du moniteur de déclenchement
- Commandes IBM MQSC
  - RUNMQSC, Exécution des commandes IBM MQSC
  - STRMQMMQSC, commandes Démarrer IBM MQSC

Groupe 2

Le reste des commandes, pour lesquelles deux niveaux de droits sont requis:

1. Droits IBM i pour l'exécution de la commande. Un administrateur IBM MQ le définit à l'aide de la commande **GRTOBJAUT** pour remplacer la restriction *PUBLIC (*EXCLUDE) pour un utilisateur ou un groupe d'utilisateurs.

   Exemple :

   ```
   GRTOBJAUT OBJ(QMQM/DSPMQMQ) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
   ```

2. Droits IBM MQ permettant de manipuler les objets IBM MQ associés à la ou aux commandes, avec les droits IBM i appropriés à l'étape 1.

   Ces droits sont contrôlés par l'utilisateur disposant des droits OAM appropriés pour l'action requise, définis par un administrateur IBM MQ à l'aide de la commande **GRTMQMAUT**

   Exemple :

   ```
   GRTMQMAUT *connect authority to the queue manager + *admchg authority to
           the queue
   ```

Les commandes peuvent être regroupées comme suit:

- Commandes relatives aux canaux
  - CHGMQMCHL, modification du canal IBM MQ

    Cela nécessite des droits de connexion * au gestionnaire de files d'attente et des droits d'accès * admchg au canal.
  - CPYMQMCHL, Copie du canal IBM MQ

    Pour cela, vous devez disposer des droits * connect et * admcrt sur le gestionnaire de files d'attente, des droits * admdsp sur le type de canal par défaut à copier et des droits * admcrt sur la classe d'objets du canal.

    Par exemple, la copie d'un canal émetteur nécessite le droit * admdsp sur SYSTEM.DEF.SENDER DEF.SENDER
  - CRTMQMCHL, Créer un canal IBM MQ

    Pour cela, vous devez disposer des droits * connect et * admcrt sur le gestionnaire de files d'attente, * admdsp sur le type de canal par défaut à créer et * admcrt sur la classe d'objets du canal.

    Par exemple, la création d'un canal émetteur requiert le droit * admdsp sur SYSTEM.DEF.SENDER DEF.SENDER
  - DLTMQMCHL, Supprimer le canal IBM MQ

    Cela nécessite * le droit de connexion au gestionnaire de files d'attente et * le droit d'admdlt au canal.
  - RSVMQMCHL, Résolution du canal IBM MQ

    Cela nécessite * l'autorisation de connexion au gestionnaire de files d'attente et * l'autorisation ctrlx au canal.
- Afficher les commandes

  Pour traiter les commandes DSP, vous devez accorder à l'utilisateur les droits `*connect` et `*admdsp` sur le gestionnaire de files d'attente, ainsi que toute option spécifique répertoriée:
  - DSPMQM, Affichage du gestionnaire de files d'attente de messages
  - DSPMQMAUT, Affichage des droits sur les objets IBM MQ
  - DSPMQMAUTI, Affichage des IBM MQ informations d'authentification- `*admdsp` à l'objet d'informations d'authentification
  - DSPMQMCHL, Affichage du canal IBM MQ - `*admdsp` vers le canal
  - DSPMQMCSVR, Affichage du serveur de commandes IBM MQ
  - DSPMQMNL, Affichage de la IBM MQ liste de noms- `*admdsp` dans la liste de noms
  - DSPMQMOBJN, Affichage des noms d'objet IBM MQ
  - DSPMQMPRC, Affichage du processus IBM MQ - `*admdsp` au processus
  - DSPMQMQ, Affichage de la file d'attente IBM MQ - `*admdsp` dans la file d'attente
  - DSPMQMTOP, Affichage de la IBM MQ rubrique- `*admdsp` à la rubrique
- Gestion des commandes

  Pour traiter les commandes WRK et afficher le panneau des options, vous devez accorder à l'utilisateur les droits `*connect` et `*admdsp` sur le gestionnaire de files d'attente, ainsi que toute option spécifique répertoriée:
  - WRKMQM, Gestion des gestionnaires de files d'attente de messages
  - WRKMQMAUT, Gestion des droits sur les objets IBM MQ
  - WRKMQMAUTD, Gestion des données de droits sur les objets IBM MQ
  - WRKMQMAUTI, Utilisation des informations d'authentification IBM MQ

- *admchg pour la commande Change IBM MQ Authentication Information Object.
- *admcrt pour la commande Create and Copy IBM MQ Authentication Information Object.
- *admdlt pour la commande Delete IBM MQ Authentication Information Object.
- *admdsp pour la commande Display IBM MQ Authentication Information Object.

– WRKMQMCHL, Utiliser le canal IBM MQ

  Pour cela, vous devez disposer des droits suivants:

  - *admchg pour la commande Change IBM MQ Channel.
  - *admclr pour la commande Clear IBM MQ Channel.
  - *admcrt pour la commande Créer et copier IBM MQ Channel.
  - *admdlt pour la commande Delete IBM MQ Channel.
  - *admdsp pour la commande Display IBM MQ Channel.
  - *ctrl pour la commande Start IBM MQ Channel.
  - *ctrl pour la commande End IBM MQ Channel.
  - *ctrl pour la commande de canal IBM MQ Ping.
  - *ctrlx pour la commande Reset IBM MQ Channel.
  - *ctrlx pour la commande Resolve IBM MQ Channel.

– WRKMQMTCSPS, Gestion de l'état des canaux IBM MQ

  Pour cela, vous devez disposer des droits *admdsp sur le canal.

– WRKMQMCL, Gestion des clusters IBM MQ
– WRKMQMCLQ, Gestion des files d'attente de cluster IBM MQ
– WRKMQMCLQM, Utilisation du gestionnaire de files d'attente de cluster IBM MQ
– WRKMQMLSR, Utilisation du programme d'écoute IBM MQ
– WRKMQMMSG, Gestion des messages IBM MQ

  Pour cela, vous devez disposer des droits *browse sur la file d'attente.

– WRKMQMNL, Gestion des listes de noms IBM MQ

  Pour cela, vous devez disposer des droits suivants:

  - *admchg pour la commande Change IBM MQ Namelist.
  - *admcrt pour la commande Create and Copy IBM MQ Namelist.
  - *admdlt pour la commande Delete IBM MQ Namelist.
  - *admdsp pour la commande Display IBM MQ Namelist.

– WRKMQMPRC, Gestion des processus IBM MQ

  Pour cela, vous devez disposer des droits suivants:

  - *admchg pour la commande Change IBM MQ Process.
  - *admcrt pour la commande Créer et copier IBM MQ Process.
  - *admdlt pour la commande Delete IBM MQ Process.
  - *admdsp pour la commande Display IBM MQ Process.

– WRKMQMQ, Gestion des files d'attente IBM MQ

  Pour cela, vous devez disposer des droits suivants:

  - *admchg pour la commande Modifier la file d'attente IBM MQ .
  - *admclr pour la commande Clear IBM MQ Queue.
  - *admcrt pour la commande Create and Copy IBM MQ Queue.
  - *admdlt pour la commande Delete IBM MQ Queue.

- *admdsp pour la commande Display IBM MQ Queue.

– WRKMQMQSTS, Gestion de l'état de la file d'attente IBM MQ

– WRKMQMTOP, Gestion des rubriques IBM MQ

  Pour cela, vous devez disposer des droits suivants:

  - *admchg pour la commande Change IBM MQ Topic.
  - *admcrt pour la commande Create and Copy IBM MQ Topic.
  - *admdlt pour la commande Delete IBM MQ Topic.
  - *admdsp pour la commande Display IBM MQ Topic.

– WRKMQMSUB, Gestion des abonnements IBM MQ

• Autres commandes de canal

  Pour traiter les commandes de canal, vous devez accorder à l'utilisateur les droits spécifiques répertoriés:

  – ENDMQMCHL, Arrêt du canal IBM MQ

    Cela nécessite des droits *connect sur le gestionnaire de files d'attente et des droits *allmqi sur la file d'attente de transmission associée au canal.

  – ENDMQMLSR, Fin du programme d'écoute IBM MQ

    Cela nécessite des droits *connect sur le gestionnaire de files d'attente et des droits *ctrl sur l'objet programme d'écoute nommé.

  – PNGMQMCHL, canal IBM MQ Ping

    Pour cela, vous devez disposer des droits *connect et *inq sur le gestionnaire de files d'attente et des droits *ctrl sur l'objet canal.

  – RSTMQMCHL, réinitialisation du canal IBM MQ

    Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

  – STRMQMCHL, démarrage du canal IBM MQ

    Cela nécessite des droits *connect sur le gestionnaire de files d'attente et des droits *ctrl sur l'objet canal.

  – STRMQMCHLI, démarrage de l'initialisateur de canal IBM MQ

    Cela requiert des droits *connect et *inq sur le gestionnaire de files d'attente et des droits *allmqi sur la file d'attente d'initialisation associée à la file d'attente de transmission du canal.

  – STRMQMLSR, démarrage du programme d'écoute IBM MQ

    Pour cela, vous devez disposer du droit de connexion * au gestionnaire de files d'attente et du droit de contrôle * ctrl sur l'objet programme d'écoute nommé.

• Autres commandes:

  Pour traiter les commandes suivantes, vous devez accorder à l'utilisateur les droits spécifiques répertoriés:

  – CCTMQM, Connexion au gestionnaire de files d'attente de messages

    Cela ne nécessite aucun droit sur les objets IBM MQ .

  – CHGMQM, modification du gestionnaire de files d'attente de messages

    Pour cela, vous devez disposer des droits *connect et *admchg sur le gestionnaire de files d'attente.

  – CHGMQMAUTI, modification des informations d'authentification IBM MQ

    Cela requiert des droits *connect sur le gestionnaire de files d'attente et des droits *admchg et *admdsp sur l'objet d'informations d'authentification.

  – CHGMQMNL, modification de la liste de noms IBM MQ

Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admchg* sur la liste de noms.

- CHGMQMPRC, modification du processus IBM MQ

  Cela requiert le droit *connect* sur le gestionnaire de files d'attente et le droit *admchg* sur le processus.

- CHGMQMQ, Modification de la file d'attente IBM MQ

  Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admchg* sur la file d'attente.

- CLRMQMQ, Effacement de la file d'attente IBM MQ

  Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admclr* sur la file d'attente.

- CPYMQMAUTI, Copie des informations d'authentification IBM MQ

  Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admdsp* sur l'objet d'informations d'authentification et des droits *admcrt* sur la classe d'objets d'informations d'authentification.

- CPYMQMNL, Copie de la liste de noms IBM MQ

  Pour cela, vous devez disposer des droits *connect* et *admcrt* sur le gestionnaire de files d'attente.

- CPYMQMPRC, Copie du processus IBM MQ

  Pour cela, vous devez disposer des droits *connect* et *admcrt* sur le gestionnaire de files d'attente.

- CPYMQMQ, Copie de la file d'attente IBM MQ

  Pour cela, vous devez disposer des droits *connect* et *admcrt* sur le gestionnaire de files d'attente.

- CRTMQMAUTI, Création des informations d'authentification IBM MQ

  Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admdsp* sur l'objet d'informations d'authentification et des droits *admcrt* sur la classe d'objets d'informations d'authentification.

- CRTMQMNL, Création d'une liste de noms IBM MQ

  Cela requiert des droits *connect* et *admcrt* sur le gestionnaire de files d'attente et des droits *admdsp* sur la liste de noms par défaut.

- CRTMQMPRC, création d'un processus IBM MQ

  Cela nécessite des droits *connect* et *admcrt* sur le gestionnaire de files d'attente et des droits *admdsp* sur le processus par défaut.

- CRTMQMQ, Création d'une file d'attente IBM MQ

  Cela nécessite des droits *connect* et *admcrt* sur le gestionnaire de files d'attente et des droits *admdsp* sur la file d'attente par défaut.

- CVTMQMDTA, Commande de conversion de type de données IBM MQ

  Cela ne nécessite aucun droit sur les objets IBM MQ .

- DLTMQMAUTI, Suppression des informations d'authentification IBM MQ

  Cela requiert des droits *connect* sur le gestionnaire de files d'attente et des droits *ctrlx* sur l'objet d'informations d'authentification.

- DLTMQMNL, Suppression de la liste de noms IBM MQ

  Cela nécessite des droits *connect* sur le gestionnaire de files d'attente et des droits *admdlt* sur la liste de noms.

- DLTMQMPRC, Suppression du processus IBM MQ

Cela requiert le droit *connect sur le gestionnaire de files d'attente et le droit *admdlt sur le processus.

- DLTMQMQ, Suppression de la file d'attente IBM MQ

  Cela nécessite des droits *connect sur le gestionnaire de files d'attente et des droits *admdlt sur la file d'attente.

- DSCMQM, Déconnexion du gestionnaire de files d'attente de messages

  Cela ne nécessite aucun droit sur les objets IBM MQ .

- RFRMQMAUT, Actualiser la sécurité

  Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

- RFRMQMCL, Actualiser le cluster

  Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

- RSMMQMCLQM, Reprise du gestionnaire de files d'attente de cluster

  Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

- RSTMQMCL, réinitialisation du cluster

  Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

- SPDMQMCLQM, Interrompre le gestionnaire de files d'attente de cluster

  Pour cela, vous devez disposer des droits *connect sur le gestionnaire de files d'attente.

IBM i *Autorisations d'accès sur IBM i*

Utilisez ces informations pour comprendre les commandes d'autorisation d'accès.

Les autorisations définies par le mot clé AUT sur les commandes GRTMQMAUT et RVKMQMAUT peuvent être catégorisées comme suit:

- Autorisations liées aux appels MQI
- Commandes d'administration liées aux autorisations
- Autorisations de contexte
- Autorisations générales, c'est-à-dire pour les appels MQI, pour les commandes ou les deux

Les tableaux suivants répertorient les différents droits, à l'aide du paramètre AUT pour les appels MQI, les appels de contexte, les commandes MQSC et PCF et les opérations génériques.

Tableau 15. Autorisations pour les appels MQI

AUT	Description
*ALTUSR	Autorisez l'utilisation des droits d'un autre utilisateur pour les appels MQOPEN et MQPUT1 .
*BROWSE	Extrayez un message d'une file d'attente en émettant un appel MQGET avec l'option BROWSE.
*CONNECT	Connectez l'application au gestionnaire de files d'attente spécifié en émettant un appel MQCONN.
*GET	Extrayez un message d'une file d'attente en émettant un appel MQGET.
*INQ	Effectuez une interrogation sur une file d'attente spécifique en émettant un appel MQINQ.
*PUB	Ouvrez une rubrique pour publier un message à l'aide d'un appel MQPUT.
*PUT	Insérez un message dans une file d'attente spécifique en émettant un appel MQPUT.

Tableau 15. Autorisations pour les appels MQI (suite)	
AUT	**Description**
*RESUME	Reprenez un abonnement à l'aide d'un appel MQSUB.
*SET	Définissez les attributs d'une file d'attente à partir de l'interface MQI en émettant un appel MQSET. Si vous ouvrez une file d'attente pour plusieurs options, vous devez être autorisé pour chacune d'elles.
*SUB	Créer, modifier ou reprendre un abonnement à une rubrique à l'aide d'un appel MQSUB.

Tableau 16. Autorisations pour les appels de contexte	
AUT	**Description**
*PASSALL	Transmettez tout le contexte à la file d'attente spécifiée. Toutes les zones de contexte sont copiées à partir de la demande d'origine.
*PASSID	Transmettez le contexte d'identité dans la file d'attente spécifiée. Le contexte d'identité est le même que celui de la demande.
*SETALL	Définit tous les contextes dans la file d'attente spécifiée. Il est utilisé par des utilitaires système spéciaux.
*SETID	Définit le contexte d'identité sur la file d'attente spécifiée. Il est utilisé par des utilitaires système spéciaux.

Tableau 17. Autorisations pour les appels MQSC et PCF	
AUT	**Description**
*ADMCHG	Modifiez les attributs de l'objet indiqué.
*ADMCLR	Mettez à blanc l'objet indiqué (commande PCF Mettre à blanc l'objet uniquement).
*ADMCRT	Créer des objets du type spécifié.
*ADMDLT	Supprimez l'objet spécifié.
*ADMDSP	Affiche les attributs de l'objet spécifié.

Tableau 18. Autorisations pour les opérations génériques	
AUT	**Description**
*ALL	Utilisez toutes les opérations applicables à l'objet. Les droits `all` sont équivalents à l'union des droits `alladm`, `allmqi`et `system` appropriés au type d'objet.
*ALLADM	Effectuez toutes les opérations d'administration applicables à l'objet.
*ALLMQI	Utilisez tous les appels MQI applicables à l'objet.
*CTRL	Contrôle le démarrage et l'arrêt des canaux, des programmes d'écoute et des services.
*CTRLX	Réinitialisez le numéro de séquence et résolvez les canaux en attente de validation.

IBM i *Utilisation des commandes d'autorisation d'accès sous IBM i*

Utilisez ces informations pour en savoir plus sur les commandes d'autorisation d'accès et utilisez les exemples de commande.

Utilisation de la commande GRTMQMAUT

Si vous disposez des droits requis, vous pouvez utiliser la commande GRTMQMAUT pour accorder à un profil utilisateur ou à un groupe d'utilisateurs l'autorisation d'accéder à un objet particulier. Les exemples suivants illustrent l'utilisation de la commande GRTMQMAUT :

1. ```
   GRTMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
             AUT(*BROWSE *PUT) MQMNAME('saturn.queue.manager')
   ```

   Dans cet exemple :

   - RED.LOCAL.QUEUE est le nom de l'objet.
   - *LCLQ (file d'attente locale) est le type d'objet.
   - GROUPA est le nom d'un profil utilisateur sur le système pour lequel les autorisations doivent être modifiées. Ce profil peut être utilisé comme profil de groupe pour d'autres utilisateurs.
   - *BROWSE et *PUT sont les autorisations accordées à la file d'attente spécifiée.

     *BROWSE ajoute l'autorisation de parcourir les messages dans la file d'attente (pour émettre une commande MQGET avec l'option de navigation).

     *PUT ajoute une autorisation d'insertion (MQPUT) de messages dans la file d'attente.
   - saturn.queue.manager est le nom du gestionnaire de files d'attente.
2. La commande suivante accorde aux utilisateurs JACK et JILL toutes les autorisations applicables, à toutes les définitions de processus, pour le gestionnaire de files d'attente par défaut.

   ```
   GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER(JACK JILL) AUT(*ALL)
   ```
3. La commande suivante accorde à l'utilisateur GEORGE le droit d'insérer un message dans la file d'attente ORDERS, sur le gestionnaire de files d'attente TRENT.

   ```
   GRTMQMAUT OBJ(TRENT) OBJTYPE(*MQM) USER(GEORGE) AUT(*CONNECT) MQMNAME (TRENT)
   GRTMQMAUT OBJ(ORDERS) OBJTYPE(*Q) USER(GEORGE) AUT(*PUT) MQMNAME (TRENT)
   ```

Utilisation de la commande RVKMQMAUT

Si vous disposez de l'autorisation requise, vous pouvez utiliser la commande RVKMQMAUT pour supprimer l'autorisation accordée précédemment à un profil utilisateur ou à un groupe d'utilisateurs pour accéder à un objet particulier. Les exemples suivants illustrent l'utilisation de la commande RVKMQMAUT :

1. ```
   RVKMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
   AUT(*PUT) MQMNAME('saturn.queue.manager')
   ```

   Les droits d'insertion de messages dans la file d'attente spécifiée, qui ont été accordés dans l'exemple précédent, sont supprimés pour GROUPA.
2. ```
   RVKMQMAUT OBJ(PAY*) OBJTYPE(*Q) USER(*PUBLIC) AUT(*GET) +
   MQMNAME(PAYROLLQM)
   ```

   Le droit d'obtenir des messages à partir de n'importe quelle file d'attente dont le nom commence par les caractères PAY, appartenant au gestionnaire de files d'attente PAYROLLQM, est supprimé de tous les utilisateurs du système, sauf s'ils, ou un groupe auquel ils appartiennent, ont été autorisés séparément.
   ```

Utilisation de la commande DSPMQMAUT

L'affichage des droits MQM (DSPMQMAUT) affiche, pour l'objet et l'utilisateur spécifiés, la liste des autorisations dont dispose l'utilisateur pour l'objet. L'exemple suivant illustre l'utilisation de la commande:

```
DSPMQMAUT OBJ(ADMINNL) OBJTYPE(*NMLIST) USER(JOE) OUTPUT(*PRINT) +
MQMNAME(ADMINQM)
```

Utilisation de la commande RFRMQMAUT

L'actualisation de la sécurité MQM (RFRMQMAUT) vous permet de mettre à jour immédiatement les informations du groupe d'autorisations de la méthode d'accès aux objets (OAM), en reflétant les modifications apportées au niveau du système d'exploitation, sans qu'il soit nécessaire d'arrêter et de redémarrer le gestionnaire de files d'attente. L'exemple suivant illustre l'utilisation de la commande:

```
RFRMQMAUT MQMNAME(ADMINQM)
```

IBM i Tables de spécifications d'autorisation sous IBM i

Utilisez ces informations pour déterminer l'autorisation requise pour utiliser des appels d'API particuliers, ainsi que des options spécifiques de ces appels, sur des objets de file d'attente, des objets de processus et des objets de gestionnaire de files d'attente.

Les tables de spécification d'autorisation démarrant dans Tableau 19, à la page 169 définissent précisément le fonctionnement des autorisations et les restrictions qui s'appliquent. Les tableaux s'appliquent aux situations suivantes:

- Applications qui émettent des appels MQI
- Programmes d'administration qui émettent des commandes MQSC sous forme de fichiers PCF d'échappement
- Programmes d'administration qui émettent des commandes PCF

Dans cette section, les informations sont présentées sous la forme d'un ensemble de tables qui spécifient les données suivantes:

Action à exécuter
Option MQI, commande MQSC ou commande PCF.

Objet de contrôle d'accès
File d'attente, définition de processus, gestionnaire de files d'attente, liste de noms, canal, canal de connexion client, programme d'écoute, service ou objet d'informations d'authentification.

Autorisation requise
Exprimée sous la forme d'une constante MQZAO_.

Dans les tableaux, les constantes préfixées par MQZAO_ correspondent aux mots clés de la liste d'autorisation pour les commandes **GRTMQMAUT** et **RVKMQMAUT** pour l'entité particulière. Par exemple, MQZAO_BROWSE correspond au mot clé *BROWSE ; De même, le mot clé MQZAO_SET_ALL_CONTEXT correspond au mot clé *SETALL, etc. Ces constantes sont définies dans le fichier d'en-tête cmqzc.h, qui est fourni avec le produit.

Autorisations MQI

Une application est autorisée à émettre des appels et des options MQI spécifiques uniquement si l'identificateur utilisateur sous lequel elle s'exécute (ou dont elle peut assumer les autorisations) a reçu l'autorisation appropriée.

Quatre appels MQI requièrent des vérifications d'autorisation: MQCONN, MQOPEN, MQPUT1et MQCLOSE.

Pour MQOPEN et MQPUT1, la vérification des droits est effectuée sur le nom de l'objet en cours d'ouverture et non sur le ou les noms, ce qui se produit après la résolution d'un nom. Par exemple, une application peut être autorisée à ouvrir une file d'attente alias sans avoir le droit d'ouvrir la file d'attente de base dans laquelle l'alias est résolu. La règle est que la vérification est effectuée sur la première définition rencontrée lors du processus de résolution de nom qui n'est pas un alias de gestionnaire de files d'attente, sauf si la définition d'alias de gestionnaire de files d'attente est ouverte directement ; c'est-à-dire que son nom apparaît dans la zone *ObjectName* du descripteur d'objet. Des droits sont toujours nécessaires pour l'objet en cours d'ouverture ; dans certains cas, des droits supplémentaires indépendants de la file d'attente, obtenus via une autorisation pour l'objet gestionnaire de files d'attente, sont requis.

Tableau 19, à la page 169, Tableau 20, à la page 169, Tableau 21, à la page 170et Tableau 22, à la page 170 récapitulent les autorisations requises pour chaque appel.

Remarque : Ces tables ne mentionnent pas les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services ou les objets d'informations d'authentification. En effet, aucune des autorisations ne s'applique à ces objets, à l'exception de MQOO_INQUIRE, pour lequel les mêmes autorisations s'appliquent que pour les autres objets.

Tableau 19. Autorisation de sécurité requise pour les appels MQCONN

Autorisation requise pour:	**Objet de file d'attente («1», à la page 170)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQCONN, option	Non applicable	Non applicable	MQZAO_CONNECT

Tableau 20. Autorisation de sécurité requise pour les appels MQOPEN

Autorisation requise pour:	**Objet de file d'attente («1», à la page 170)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQOO_INTERROGATION	MQZAO_INQUIRE **(«2», à la page 171)**	MQZAO_INQUIRE **(«2», à la page 171)**	MQZAO_INQUIRE **(«2», à la page 171)**
MQOO_BROWSE	MQZAO_PARCOURIR	Non applicable	Aucun contrôle
MQOO_ENTRÉE_ *	MQZAO_ENTREE	Non applicable	Aucun contrôle
MQOO_SAVE_ ALL_CONTEXT **(«3», à la page 171)**	MQZAO_ENTREE	Non applicable	Non applicable
MQOO_OUTPUT (file d'attente normale) **(«4», à la page 171)**	MQZAO_OUTPUT	Non applicable	Non applicable
MQOO_PASS_IDENTITY_CONTEXT **(«5», à la page 171)**	MQZAO_PASS_ IDENTITY_CONTEXT	Non applicable	Aucun contrôle
MQOO_PASS_ALL_ CONTEXT **(«5», à la page 171, «6», à la page 171)**	MQZAO_PASS _ALL_CONTEXT	Non applicable	Aucun contrôle
MQOO_SET_IDENTITY_CONTEXT **(«5», à la page 171, «6», à la page 171)**	MQZAO_SET_IDENTITY_CONTEXT	Non applicable	MQZAO_SET_ IDENTITY_CONTEXT **(«7», à la page 171)**
MQOO_SET_ ALL_CONTEXT **(«5», à la page 171, «8», à la page 171)**	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ ALL_CONTEXT **(«7», à la page 171)**

Tableau 20. Autorisation de sécurité requise pour les appels MQOPEN (suite)			
Autorisation requise pour:	**Objet de file d'attente («1», à la page 170)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQOO_OUTPUT (file d'attente de transmission) **(«9», à la page 171)**	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ALL_CONTEXT **(«7», à la page 171)**
MQOO_SET	MQZAO_SET	Non applicable	Aucun contrôle
MQOO_ALTERNATE_AUTORITE UTILISATEUR	**(«10», à la page 171)**	**(«10», à la page 171)**	MQZAO_ALTERNATE_USER_AUTHORITY **(«10», à la page 171, «11», à la page 171)**

Tableau 21. Autorisation de sécurité requise pour les appels MQPUT1			
Autorisation requise pour:	**Objet de file d'attente («1», à la page 170)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQPMO_PASS_IDENTITY_CONTEXT	MQZAO_PASS_IDENTITY_CONTEXT **(«12», à la page 171)**	Non applicable	Aucun contrôle
MQPMO_PASS_ALL_CONTEXT	MQZAO_PASS_ALL_CONTEXT **(«12», à la page 171)**	Non applicable	Aucun contrôle
MQPMO_SET_IDENTITY_CONTEXT	MQZAO_SET_IDENTITY_CONTEXT **(«12», à la page 171)**	Non applicable	MQZAO_SET_IDENTITY_CONTEXT **(«7», à la page 171)**
MQPMO_SET_ALL_CONTEXT	MQZAO_SET_ALL_CONTEXT **(«12», à la page 171)**	Non applicable	MQZAO_SET_ALL_CONTEXT **(«7», à la page 171)**
(File d'attente de transmission) **(«9», à la page 171)**	MQZAO_SET_ALL_CONTEXT	Non applicable	MQZAO_SET_ALL_CONTEXT **(«7», à la page 171)**
MQPMO_ALTERNATE_USER_AUTHORITY	**(«13», à la page 171)**	Non applicable	MQZAO_ALTERNATE_USER_AUTHORITY **(«11», à la page 171)**

Tableau 22. Autorisation de sécurité requise pour les appels MQCLOSE			
Autorisation requise pour:	**Objet de file d'attente («1», à la page 170)**	**Objet Processus**	**Objet gestionnaire de files d'attente**
MQCO_DELETE	MQZAO_DELETE **(«14», à la page 171)**	Non applicable	Non applicable
MQCO_DELETE _PURGE	MQZAO_DELETE **(«14», à la page 171)**	Non applicable	Non applicable

Remarques relatives aux tableaux:

1. Si une file d'attente modèle est ouverte:
   - Le droit MQZAO_DISPLAY est requis pour la file d'attente modèle, en plus du droit d'ouverture de la file d'attente modèle pour le type d'accès pour lequel vous l'ouvrez.
   - Les droits MQZAO_CREATE ne sont pas nécessaires pour créer la file d'attente dynamique.

- L'ID utilisateur utilisé pour ouvrir la file d'attente modèle reçoit automatiquement tous les droits spécifiques à la file d'attente (équivalents à MQZAO_ALL) pour la file d'attente dynamique créée.

2. L'objet file d'attente, processus, liste de noms ou gestionnaire de files d'attente est vérifié, en fonction du type d'objet ouvert.
3. MQOO_INPUT_ * doit également être spécifié. Cette option est valide pour une file d'attente locale, modèle ou alias.
4. Cette vérification est effectuée pour toutes les observations de sortie, à l'exception de la casse spécifiée dans la remarque «9», à la page 171.
5. MQOO_OUTPUT doit également être spécifié.
6. MQOO_PASS_IDENTITY_CONTEXT est également impliqué par cette option.
7. Ce droit est requis pour l'objet gestionnaire de files d'attente et la file d'attente particulière.
8. MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT et MQOO_SET_IDENTITY_CONTEXT sont également impliquées par cette option.
9. Cette vérification est effectuée pour une file d'attente locale ou modèle dont l'attribut de file d'attente *Utilisation* est MQUS_TRANSMISSION et qui est ouverte directement pour la sortie. Elle ne s'applique pas si une file d'attente éloignée est ouverte (soit en spécifiant les noms du gestionnaire de files d'attente éloignées et de la file d'attente éloignée, soit en indiquant le nom d'une définition locale de la file d'attente éloignée).
10. Au moins l'une des options MQOO_INQUIRE (pour tout type d'objet) ou (pour les files d'attente) MQOO_BROWSE, MQOO_INPUT_ *, MQOO_OUTPUT ou MQOO_SET doit également être spécifiée. La vérification effectuée est la même que pour les autres options spécifiées, à l'aide de l'identificateur d'utilisateur de remplacement fourni pour les droits sur les objets nommés spécifiques, et des droits d'application en cours pour la vérification MQZAO_ALTERNATE_USER_IDENTIFIER.
11. Cette autorisation permet de spécifier tout ID *AlternateUser* .
12. Une vérification MQZAO_OUTPUT est également effectuée si la file d'attente ne possède pas l'attribut de file d'attente *Usage* MQUS_TRANSMISSION.
13. La vérification effectuée est la même que pour les autres options spécifiées, à l'aide de l'identificateur d'utilisateur de remplacement fourni pour l'autorité de file d'attente nommée et de l'autorité d'application en cours pour la vérification MQZAO_ALTERNATE_USER_IDENTIFIER.
14. La vérification est effectuée uniquement si les deux affirmations suivantes sont vraies:
    - Une file d'attente dynamique permanente est en cours de fermeture et de suppression.
    - La file d'attente n'a pas été créée par le MQOPEN qui a renvoyé le descripteur d'objet utilisé.

    Sinon, il n'y a pas de contrôle.

Remarques générales:

1. L'autorisation spéciale MQZAO_ALL_MQI inclut toutes les autorisations suivantes qui sont pertinentes pour le type d'objet:
   - MQZAO_CONNECT
   - MQZAO_INQUIRE
   - MQZAO_SET
   - MQZAO_PARCOURIR
   - MQZAO_ENTREE
   - MQZAO_OUTPUT
   - MQZAO_PASS_IDENTITY_CONTEXT
   - MQZAO_PASS_ALL_CONTEXT
   - MQZAO_SET_IDENTITY_CONTEXT
   - MQZAO_SET_ALL_CONTEXT
   - MQZAO_ALTERNATE_USER_AUTHORITY

2. MQZAO_DELETE (voir la remarque «14», à la page 171) et MQZAO_DISPLAY sont classés en tant qu'autorisations d'administration. Ils ne sont donc pas inclus dans MQZAO_ALL_MQI.
3. *Aucune vérification* signifie qu'aucune vérification d'autorisation n'est effectuée.
4. *Non applicable* signifie que le contrôle d'autorisation n'est pas pertinent pour cette opération. Par exemple, vous ne pouvez pas émettre un appel MQPUT à un objet de processus.

IBM i *Autorisations pour les commandes MQSC dans les fichiers PCF d'échappement sous IBM i*

Ces autorisations permettent à un utilisateur d'émettre des commandes d'administration en tant que message PCF d'arrêt programme. Ces méthodes permettent à un programme d'envoyer une commande d'administration sous forme de message à un gestionnaire de files d'attente, pour exécution pour le compte de cet utilisateur.

Cette section récapitule les autorisations nécessaires pour chaque commande MQSC contenue dans Escape PCF.

Non applicable signifie que le contrôle d'autorisation n'est pas pertinent pour cette opération.

L'ID utilisateur sous lequel le programme qui soumet la commande s'exécute doit également disposer des droits suivants:

- Droits MQZAO_CONNECT sur le gestionnaire de files d'attente
- Droits DISPLAY sur le gestionnaire de files d'attente afin d'exécuter des commandes PCF
- Droit d'émettre les commandes MQSC dans le texte de la commande Escape PCF

ALTER *objet*

Objet	**Autorisation requise**
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

CLEAR *objet*

Objet	**Autorisation requise**
File d'attente	MQZAO_CLEAR
Topic	MQZAO_CLEAR
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable

Objet	Autorisation requise
Canal	Non applicable
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

DEFINE *objet* NOREPLACE («1», à la page 176)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(«2», à la page 176)**
Topic	MQZAO_CREATE **(«2», à la page 176)**
Processus	MQZAO_CREATE **(«2», à la page 176)**
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_CREATE **(«2», à la page 176)**
Informations d'authentification	MQZAO_CREATE **(«2», à la page 176)**
Canal	MQZAO_CREATE **(«2», à la page 176)**
Canal de connexion client	MQZAO_CREATE **(«2», à la page 176)**
Programme d'écoute	MQZAO_CREATE **(«2», à la page 176)**
Service	MQZAO_CREATE **(«2», à la page 176)**

DEFINE *objet* REPLACE («1», à la page 176, «3», à la page 176)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

DELETE *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DELETE
Topic	MQZAO_DELETE
Processus	MQZAO_DELETE
Gestionnaire de files d'attente	Non applicable

Objet	Autorisation requise
Liste de noms	MQZAO_DELETE
Informations d'authentification	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de connexion client	MQZAO_DELETE
Programme d'écoute	MQZAO_DELETE
Service	MQZAO_DELETE

DISPLAY *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DISPLAY
Topic	MQZAO_DISPLAY
Processus	MQZAO_DISPLAY
Gestionnaire de files d'attente	MQZAO_DISPLAY
Liste de noms	MQZAO_DISPLAY
Informations d'authentification	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de connexion client	MQZAO_DISPLAY
Programme d'écoute	
Service	

Envoyer une commande PING à un canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Réinitialisation du canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable

Objet	Autorisation requise
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	MQZAO_CONTRÔLE_ÉTENDU
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Résolution du canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	MQZAO_CONTRÔLE_ÉTENDU
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

START *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ

STOP *objet*

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	CONTROLE MQZ
Service	CONTROLE MQZ

Remarque :

1. Pour les commandes DEFINE, le droit MQZAO_DISPLAY est également requis pour l'objet LIKE si un tel droit est spécifié, ou sur le système SYSTEM.DEFAULT.xxx si LIKE est omis.
2. Les droits MQZAO_CREATE ne sont pas spécifiques à un objet ou à un type d'objet particulier. Le droit de création est accordé pour tous les objets d'un gestionnaire de files d'attente spécifié, en spécifiant un type d'objet QMGR dans la commande GRTMQMAUT .
3. Cette option s'applique si l'objet à remplacer existe déjà. Si tel n'est pas le cas, la vérification est celle de l'objet DEFINE NOREPLACE.

IBM i *Autorisations pour les commandes PCF sous IBM i*

Ces autorisations permettent à un utilisateur d'émettre des commandes d'administration en tant que commandes PCF. Ces méthodes permettent à un programme d'envoyer une commande d'administration sous forme de message à un gestionnaire de files d'attente, pour exécution pour le compte de cet utilisateur.

Cette section récapitule les autorisations requises pour chaque commande PCF.

Aucune vérification signifie qu'aucune vérification d'autorisation n'est effectuée ; *Non applicable* signifie que la vérification d'autorisation n'est pas pertinente pour cette opération.

L'ID utilisateur sous lequel le programme qui soumet la commande s'exécute doit également disposer des droits suivants:

- Droits MQZAO_CONNECT sur le gestionnaire de files d'attente
- Droits DISPLAY sur le gestionnaire de files d'attente afin d'exécuter des commandes PCF

L'autorisation spéciale MQZAO_ALL_ADMIN inclut les autorisations suivantes:

- MODIFICATION MQZAO_DE
- MQZAO_CLEAR
- MQZAO_DELETE
- MQZAO_DISPLAY
- CONTROLE MQZ
- MQZAO_CONTRÔLE_ÉTENDU

MQZAO_CREATE n'est pas inclus car il n'est pas spécifique à un objet ou à un type d'objet particulier

Modifier *objet*

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	MODIFICATION MQZAO_DE
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

Effacer *objet*

Objet	Autorisation requise
File d'attente	MQZAO_CLEAR
Topic	MQZAO_CLEAR
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	Non applicable
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Copier l'objet (sans remplacement) («1», à la page 182)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(«2», à la page 182)**
Topic	MQZAO_CREATE **(«2», à la page 182)**
Processus	MQZAO_CREATE **(«2», à la page 182)**
Gestionnaire de files d'attente	Non applicable
NomelistMQZAO_CREATE	MQZAO_CREATE **(«2», à la page 182)**
Informations d'authentification	MQZAO_CREATE **(«2», à la page 182)**
Canal	MQZAO_CREATE **(«2», à la page 182)**
Canal de connexion client	MQZAO_CREATE **(«2», à la page 182)**
Programme d'écoute	MQZAO_CREATE **(«2», à la page 182)**

Objet	Autorisation requise
Service	MQZAO_CREATE **(«2», à la page 182)**

Copiez l'objet (avec remplacement) («1», à la page 182, «4», à la page 182)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

Créer un objet (sans remplacement) («3», à la page 182)

Objet	Autorisation requise
File d'attente	MQZAO_CREATE **(«2», à la page 182)**
Topic	MQZAO_CREATE **(«2», à la page 182)**
Processus	MQZAO_CREATE **(«2», à la page 182)**
Gestionnaire de files d'attente	Non applicable
Liste de noms	MQZAO_CREATE **(«2», à la page 182)**
Informations d'authentification	MQZAO_CREATE **(«2», à la page 182)**
Canal	MQZAO_CREATE **(«2», à la page 182)**
Canal de connexion client	MQZAO_CREATE **(«2», à la page 182)**
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

Créer un objet (avec remplacement) («3», à la page 182, «4», à la page 182)

Objet	Autorisation requise
File d'attente	MODIFICATION MQZAO_DE
Topic	MODIFICATION MQZAO_DE
Processus	MODIFICATION MQZAO_DE
Gestionnaire de files d'attente	Non applicable
Liste de noms	MODIFICATION MQZAO_DE
Informations d'authentification	MODIFICATION MQZAO_DE
Canal	MODIFICATION MQZAO_DE

Objet	Autorisation requise
Canal de connexion client	MODIFICATION MQZAO_DE
Programme d'écoute	MODIFICATION MQZAO_DE
Service	MODIFICATION MQZAO_DE

Supprimer *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DELETE
Topic	MQZAO_DELETE
Processus	MQZAO_DELETE
Gestionnaire de files d'attente	MQZAO_DELETE
Liste de noms	MQZAO_DELETE
Informations d'authentification	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de connexion client	MQZAO_DELETE
Programme d'écoute	MQZAO_DELETE
Service	MQZAO_DELETE

Interroger *objet*

Objet	Autorisation requise
File d'attente	MQZAO_DISPLAY
Topic	MQZAO_DISPLAY
Processus	MQZAO_DISPLAY
Gestionnaire de files d'attente	MQZAO_DISPLAY
Liste de noms	MQZAO_DISPLAY
Informations d'authentification	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de connexion client	MQZAO_DISPLAY
Programme d'écoute	MQZAO_DISPLAY
Service	MQZAO_DISPLAY

Consulter les noms d' *objet*

Objet	Autorisation requise
File d'attente	Aucun contrôle
Topic	Aucun contrôle
Processus	Aucun contrôle
Gestionnaire de files d'attente	Aucun contrôle
Liste de noms	Aucun contrôle

Objet	Autorisation requise
Informations d'authentification	Aucun contrôle
Canal	Aucun contrôle
Canal de connexion client	Aucun contrôle
Programme d'écoute	Aucun contrôle
Service	Aucun contrôle

Envoyer une commande PING à un canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Réinitialisation du canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	MQZAO_CONTRÔLE_ÉTENDU
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Réinitialiser les statistiques de file d'attente

Objet	Autorisation requise
File d'attente	MQZAO_DISPLAY et MQZAO_CHANGE
Topic	Non applicable
Processus	Non applicable

Objet	Autorisation requise
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	Non applicable
Canal de connexion client	Non applicable
Programme d'écoute	
Service	

Résolution du canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	MQZAO_CONTRÔLE_ÉTENDU
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Démarrer un canal

Objet	Autorisation requise
File d'attente	Non applicable
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Arrêter le canal

Objet	Autorisation requise
File d'attente	Non applicable

Objet	Autorisation requise
Topic	Non applicable
Processus	Non applicable
Gestionnaire de files d'attente	Non applicable
Liste de noms	Non applicable
Informations d'authentification	Non applicable
Canal	CONTROLE MQZ
Canal de connexion client	Non applicable
Programme d'écoute	Non applicable
Service	Non applicable

Remarque :

1. Pour les commandes de copie, le droit MQZAO_DISPLAY est également requis pour l'objet From.
2. Les droits MQZAO_CREATE ne sont pas spécifiques à un objet ou à un type d'objet particulier. Le droit de création est accordé pour tous les objets d'un gestionnaire de files d'attente spécifié, en spécifiant un type d'objet QMGR dans la commande GRTMQMAUT .
3. Pour les commandes de création, les droits MQZAO_DISPLAY sont également requis pour le système SYSTEM.DEFAULT.* .
4. Cette option s'applique si l'objet à remplacer existe déjà. Si ce n'est pas le cas, la vérification est la même que pour la copie ou la création sans remplacement.

IBM i Profils OAM génériques sous IBM i

Les profils génériques du gestionnaire des droits d'accès aux objets (OAM) vous permettent de définir les droits d'accès d'un utilisateur à de nombreux objets à la fois, au lieu d'avoir à émettre des commandes **GRTMQMAUT** distinctes pour chaque objet individuel lors de sa création. L'utilisation de profils génériques dans la commande **GRTMQMAUT** vous permet de définir un droit générique pour tous les futurs objets créés qui correspondent à ce profil.

Le reste de cette section décrit plus en détail l'utilisation des profils génériques:

- «Utilisation des caractères génériques», à la page 182
- «Priorités de profil», à la page 183

Utilisation des caractères génériques

Ce qui rend un profil générique, c'est l'utilisation de caractères spéciaux (caractères génériques) dans le nom du profil. Par exemple, le caractère générique point d'interrogation (?) correspond à n'importe quel caractère unique dans un nom. Par conséquent, si vous spécifiez ABC.?EF, l'autorisation que vous accordez à ce profil s'applique à tous les objets créés avec les noms ABC.DEF, ABC.CEF, ABC.BEF, etc.

Les caractères génériques disponibles sont les suivants:

?

Utilisez le point d'interrogation (?) au lieu de n'importe quel caractère. Par exemple, AB.?D s'applique aux objets AB.CD, AB.EDet AB.FD.

*****

Utilisez l'astérisque (*) comme suit:

- Un *qualificateur* dans un nom de profil pour correspondre à un qualificateur dans un nom d'objet. Le qualificatif est une partie de nom d'objet délimitée par un point. Par exemple, dans ABC.DEF.GHI, les qualificatifs sont ABC, DEF et GHI.

Par exemple, ABC.*.JKL s'applique aux objets ABC.DEF.JKLet ABC.GHI.JKL. (Notez que cela ne s'applique **pas** à ABC.JKL ; * utilisé dans ce contexte indique toujours un qualificateur.)

- Caractère dans un qualificatif d'un nom de profil qui doit correspondre à zéro ou plusieurs caractères dans le qualificatif d'un nom d'objet.

  Par exemple, ABC.DE*.JKL s'applique aux objets ABC.DE.JKL, ABC.DEF.JKLet ABC.DEGH.JKL.

**

Utilisez le double astérisque (**) ***une fois*** dans un nom de profil comme suit:

- Nom de profil complet correspondant à tous les noms d'objet. Par exemple, si vous utilisez le mot clé OBJTYPE (*PRC) pour identifier les processus, puis utilisez ** comme nom de profil, vous modifiez les autorisations pour tous les processus.
- Comme qualificatif de début, de milieu ou de fin dans un nom de profil pour correspondre à zéro ou plusieurs qualificatifs dans un nom d'objet. Par exemple, **.ABC identifie tous les objets avec le qualificateur final ABC.

Priorités de profil

Un point important à comprendre lors de l'utilisation de profils génériques est la priorité donnée aux profils lors de la détermination des droits à appliquer à un objet en cours de création. Par exemple, supposons que vous ayez émis les commandes suivantes:

```
GRTMQMAUT OBJ(AB.*) OBJTYPE(*Q) USER(FRED) AUT(*PUT) MQMNAME(MYQMGR)
GRTMQMAUT OBJ(AB.C*) OBJTYPE(*Q) USER(FRED) AUT(*GET) MQMNAME(MYQMGR)
```

Le premier accorde le droit d'insertion à toutes les files d'attente pour le principal FRED dont les noms correspondent au profil AB. * ; La seconde permet d'obtenir des droits sur les mêmes types de file d'attente qui correspondent au profil AB.C*.

Supposons que vous créiez maintenant une file d'attente appelée AB.CD. Selon les règles de correspondance des caractères génériques, GRTMQMAUT peut s'appliquer à cette file d'attente. Alors, a-t-elle mis ou obtenu l'autorité?

Pour trouver la réponse, vous appliquez la règle selon laquelle, chaque fois que plusieurs profils peuvent s'appliquer à un objet, **seuls les plus spécifiques s'appliquent**. La façon dont vous appliquez cette règle consiste à comparer les noms de profil de gauche à droite. Lorsqu'ils diffèrent, un caractère non générique est plus spécifique qu'un caractère générique. Ainsi, dans l'exemple précédent, la file d'attente AB.CD dispose du droit **get** (AB.C* est plus spécifique que AB. *).

Lorsque vous comparez des caractères génériques, l'ordre de la *spécificité* est le suivant:

1. ?
2. *
3. **

IBM i Spécification du service d'autorisation installé sous IBM i

Vous pouvez spécifier le composant de service d'autorisation à utiliser.

Le paramètre **Service Component name** sous **GRTMQMAUT** et **RVKMQMAUT** vous permet de spécifier le nom du composant de service d'autorisation installé.

La sélection de **F24** dans le panneau initial, suivie de **F9=All parameters** dans le panneau suivant de l'une des commandes, vous permet de spécifier le composant d'autorisation installé (*DFT) ou le nom du composant de service d'autorisation requis spécifié dans la section Service du fichier qm.ini du gestionnaire de files d'attente.

DSPMQMAUT dispose également de ce paramètre supplémentaire. Ce paramètre permet de rechercher tous les composants d'autorisation installés (*DFT), ou le nom de composant de service d'autorisation indiqué, pour le nom d'objet, le type d'objet et l'utilisateur indiqués.

IBM i Utilisation avec et sans profils de droits d'accès sous IBM i

Utilisez ces informations pour apprendre à utiliser des profils de droits d'accès et à travailler sans profils de droits d'accès.

Vous pouvez utiliser des profils d'autorité, comme expliqué dans «Utilisation des profils de droits d'accès», à la page 184, ou sans eux, comme expliqué ici:

Pour travailler sans profils de droits d'accès, utilisez *NONE comme paramètre de droits d'accès sur **GRTMQMAUT** pour créer des profils sans droits d'accès. Tous les profils existants restent inchangés.

Sous **RVKMQMAUT**, utilisez *REMOVE comme paramètre de droits d'accès pour supprimer un profil de droits d'accès existant.

Utilisation des profils de droits d'accès

Deux commandes sont associées au profilage des droits:

- **WRKMQMAUT**
- **WRKMQMAUTD**

Vous pouvez accéder à ces commandes directement à partir de la ligne de commande ou à partir du panneau WRKMQM en:

1. Entrez le nom du gestionnaire de files d'attente et appuyez sur la touche `Enter` pour accéder au panneau des résultats **WRKMQM** .
2. Sélection de `F23=More options` sur ce panneau.

L'option 24 sélectionne le panneau de résultats pour la **WRKMQMAUT** commande et l'option 25 sélectionne la commande **WRKMQMAUTI** , qui est utilisée avec la couche de liaisons SSL.

WRKMQMAUT

Cette commande permet de gérer les données de droits d'accès contenues dans la file d'attente des droits d'accès.

Remarque : Pour exécuter cette commande, vous devez disposer des droits `*connect` et `*admdsp` sur le gestionnaire de files d'attente. Toutefois, pour créer ou supprimer un profil, vous devez disposer des droits QMQMADM.

Si vous affichez les informations à l'écran, une liste des noms de profil de droits d'accès, ainsi que leurs types, s'affiche. Si vous imprimez la sortie, vous recevez une liste détaillée de toutes les données de droits d'accès, des utilisateurs enregistrés et de leurs droits d'accès.

Lorsque vous entrez un nom d'objet ou de profil dans ce panneau et que vous appuyez sur la touche Entrée, vous accédez au panneau de résultats pour **WRKMQMAUT** .

Si vous sélectionnez `4=Delete`, vous accédez à un nouveau panneau à partir duquel vous pouvez confirmer que vous souhaitez supprimer tous les noms d'utilisateur enregistrés dans le nom de profil de droits d'accès générique que vous spécifiez. Cette option exécute **RVKMQMAUT** avec l'option *REMOVE pour tous les utilisateurs et applique **uniquement** aux noms de profil génériques.

Si vous sélectionnez `12=Work with profile` , vous accédez au panneau des résultats de la commande **WRKMQMAUTD** , comme expliqué dans «WRKMQMAUTD», à la page 184.

WRKMQMAUTD

Cette commande permet d'afficher tous les utilisateurs enregistrés avec un nom de profil de droits et un type d'objet particuliers. Pour exécuter cette commande, vous devez disposer des droits `*connect` et `*admdsp` sur le gestionnaire de files d'attente. Toutefois, pour accorder, exécuter, créer ou supprimer un profil, vous devez disposer des droits QMQMADM.

La sélection de F24=More keys dans le panneau d'entrée initial, suivie de l'option F9=All Parameters , affiche le nom du composant de service comme pour **GRTMQMAUT** et **RVKMQMAUT**.

Remarque : La touche F11=Display Object Authorizations permet de basculer entre les types de droits suivants:

- Autorisations d'objet
- Autorisations de contexte
- Autorisations MQI

Les options de l'écran sont les suivantes:

2=Grant
Permet d'accéder au panneau **GRTMQMAUT** pour ajouter des droits aux droits en cours.

3=Revoke
Permet d'accéder au panneau **RVKMQMAUT** pour supprimer certaines des définitions en cours

4=Delete
Permet d'accéder à un panneau qui vous permet de supprimer les données de droits d'accès pour les utilisateurs spécifiés. **RVKMQMAUT** est exécuté avec l'option *REMOVE.

5=Display
Vous permet d'accéder à la commande **DSPMQMAUT** existante

F6=Create
Vous permet d'accéder au panneau **GRTMQMAUT** qui vous permet de créer un enregistrement de droits d'accès de profil.

IBM i Instructions relatives à Object Authority Manager sous IBM i

Conseils et astuces supplémentaires pour l'utilisation du gestionnaire des droits d'accès aux objets (OAM)

Limiter l'accès aux opérations sensibles

Certaines opérations sont sensibles ; limitez-les aux utilisateurs privilégiés. Exemple :

- Accès à certaines files d'attente spéciales, telles que les files d'attente de transmission ou la file d'attente de commandes SYSTEM.ADMIN.COMMAND.QUEUE
- Exécution de programmes qui utilisent des options de contexte MQI complètes
- Création et copie de files d'attente d'application

Répertoires du gestionnaire de files d'attente

Les répertoires et les bibliothèques contenant les files d'attente et les autres données du gestionnaire de files d'attente sont privés pour le produit. N'utilisez pas de commandes de système d'exploitation standard pour accorder ou révoquer des autorisations sur les ressources MQI.

Files d'attente

Les droits d'accès à une file d'attente dynamique sont basés sur, mais ne sont pas nécessairement les mêmes que ceux de la file d'attente modèle dont elle est dérivée.

Pour les files d'attente d'alias et les files d'attente éloignées, l'autorisation est celle de l'objet lui-même, et non celle de la file d'attente dans laquelle l'alias ou la file d'attente éloignée est résolue. Il est possible d'autoriser un profil utilisateur à accéder à une file d'attente alias qui se résout en une file d'attente locale pour laquelle le profil utilisateur ne dispose pas de droits d'accès.

Limitez les droits de création de files d'attente aux utilisateurs privilégiés. Si vous ne le faites pas, les utilisateurs peuvent ignorer le contrôle d'accès normal en créant un alias.

Droits d'utilisateur de remplacement

Les droits d'utilisateur de remplacement contrôlent si un profil utilisateur peut utiliser les droits d'un autre profil utilisateur lors de l'accès à un objet IBM MQ . Cette technique est essentielle lorsqu'un serveur reçoit des demandes d'un programme et que le serveur souhaite s'assurer que le programme dispose des droits requis pour la demande. Le serveur peut disposer des droits requis, mais il doit savoir si le programme dispose des droits requis pour les actions qu'il a demandées.

Exemple :

- Un programme serveur s'exécutant sous le profil utilisateur PAYSERV extrait un message de demande d'une file d'attente qui a été placée dans la file d'attente par le profil utilisateur USER1.
- Lorsque le programme serveur obtient le message de demande, il traite la demande et insère la réponse dans la file d'attente de réponse spécifiée dans le message de demande.
- Au lieu d'utiliser son propre profil utilisateur (PAYSERV) pour autoriser l'ouverture de la file d'attente de réponse, le serveur peut spécifier un autre profil utilisateur, dans ce cas, USER1. Dans cet exemple, vous pouvez utiliser les droits d'utilisateur de remplacement pour contrôler si PAYSERV est autorisé à spécifier USER1 comme profil d'utilisateur de remplacement lorsqu'il ouvre la file d'attente de réponse.

Le profil utilisateur de remplacement est spécifié dans la zone *AlternateUserId* du descripteur d'objet.

Remarque : Vous pouvez utiliser des profils utilisateur de remplacement sur n'importe quel objet IBM MQ . L'utilisation d'un profil utilisateur de remplacement n'affecte pas le profil utilisateur utilisé par les autres gestionnaires de ressources.

Droits d'accès au contexte

Le contexte est une information qui s'applique à un message particulier et qui est contenue dans le descripteur de message, MQMD, qui fait partie du message.

Pour la description des zones de descripteur de message relatives au contexte, voir Présentation de MQMD.

Pour plus d'informations sur les options de contexte, voir Contexte de message.

Remarques relatives à la sécurité à distance

Pour la sécurité à distance, tenez compte des points suivants:

Droit d'insertion
Pour la sécurité des gestionnaires de files d'attente, vous pouvez spécifier les droits d'insertion utilisés lorsqu'un canal reçoit un message envoyé par un autre gestionnaire de files d'attente.

Ce paramètre est valide uniquement pour les types de canal RCVR, RQSTR ou CLUSRCVR. Indiquez l'attribut de canal PUTAUT comme suit:

infrastructure d"évaluation de déploiement
Profil utilisateur par défaut. Il s'agit du profil utilisateur QMQM sous lequel l'agent MCA s'exécute.

CTX
Profil utilisateur dans le contexte de message.

Files d'attente de transmission
Les gestionnaires de files d'attente placent automatiquement les messages éloignés dans une file d'attente de transmission ; aucun droit spécial n'est requis. Toutefois, l'insertion d'un message directement dans une file d'attente de transmission nécessite une autorisation spéciale.

Exits de canal
Les exits de canal peuvent être utilisés pour une sécurité accrue.

Enregistrements d'authentification de canal
Permet d'exercer un contrôle plus précis sur l'accès accordé aux systèmes de connexion au niveau d'un canal.

Pour plus d'informations sur la sécurité à distance, voir «Autorisation de canal», à la page 111.

Protection des canaux avec SSL/TLS

Le protocole TLS (Transport Layer Security) offre une sécurité de canal, avec une protection contre les écoutes clandestines, les falsifications et les usurpations d'identité. La prise en charge de TLS par IBM MQ vous permet de spécifier, dans la définition de canal, qu'un canal particulier utilise la sécurité TLS. Vous pouvez également spécifier les détails de la sécurité de votre choix, tels que l'algorithme de chiffrement que vous souhaitez utiliser.

La prise en charge de TLS dans IBM MQ utilise l' *objet d'informations d'authentification* du gestionnaire de files d'attente et diverses commandes CL et MQSC, ainsi que des paramètres de gestionnaire de files d'attente et de canal qui définissent la prise en charge de TLS requise en détail.

Les commandes CL suivantes prennent en charge TLS:

WRKMQMAUTI
Gestion des attributs d'un objet d'informations d'authentification.

CHGMQMAUTI
Modifier les attributs d'un objet d'informations d'authentification.

CRTMQMAUTI
Créez un objet d'informations d'authentification.

CPYMQMAUTI
Créez un objet d'informations d'authentification en copiant un objet existant.

DLTMQMAUTI
Supprimez un objet d'informations d'authentification.

DSPMQMAUTI
Affiche les attributs d'un objet d'informations d'authentification spécifique.

Pour une présentation de la sécurité des canaux à l'aide de TLS, voir

- Protection des canaux avec TLS

Pour plus de détails sur les commandes PCF associées à TLS, voir

- Change, Copy, and Create Authentication Information Object
- Delete Authentication Information Object
- Inquire Authentication Information Object

z/OS Configuration de la sécurité sous z/OS

Remarques relatives à la sécurité spécifiques à z/OS.

La sécurité dans IBM MQ for z/OS est contrôlée à l'aide de RACF ou d'un gestionnaire de sécurité externe (ESM) équivalent.

Les instructions suivantes supposent que vous utilisez RACF.

Référence associée
Scénario de sécurité : Deux gestionnaire de files d'attente sous z/OS
Scénario de sécurité : deux groupes de partage de files d'attente sous z/OS

z/OS Classes de sécurité RACF

Les classes RACF sont utilisées pour contenir les profils requis pour la vérification de la sécurité IBM MQ . La plupart des classes membres ont des classes de groupe équivalentes. Vous devez activer les classes et leur permettre d'accepter les profils génériques

Chaque classe RACF contient un ou plusieurs profils utilisés à un moment donné dans la séquence de vérification, comme illustré dans la Tableau 23, à la page 188.

Tableau 23. Classes RACF utilisées par IBM MQ

Classe de membre	Classe de groupe	Contenu
MQADMIN	GMQADMIN	Profils : Utilisé principalement pour les profils de maintien pour les fonctions de type administration. Exemple : • Profils pour les commutateurs de sécurité IBM MQ • Profil de sécurité RESLEVEL • Profils pour la sécurité des utilisateurs de remplacement • Profil de sécurité de contexte • Profils pour la sécurité des ressources de commande
MXADMIN	GMXADMIN	Profils : Utilisé principalement pour les profils de maintien pour les fonctions de type administration. Exemple : • Profils pour les commutateurs de sécurité IBM MQ • Profil de sécurité RESLEVEL • Profils pour la sécurité des utilisateurs de remplacement • Profil de sécurité de contexte • Profils pour la sécurité des ressources de commande Cette classe peut contenir des profils RACF en majuscules et en casse mixte.
MQCONN		Profils utilisés pour la sécurité de connexion
MQCMDS		Profils utilisés pour la sécurité des commandes
MQQUEUE	GMQQUEUE	Profils utilisés dans la sécurité des ressources de file d'attente
MXQUEUE	GMXQUEUE	Profils en casse mixte et en majuscules utilisés dans la sécurité des ressources de file d'attente
MQPROC	GMQPROC	Profils utilisés dans la sécurité des ressources de processus
MXPROC	GMXPROC	Profils en casse mixte et en majuscules utilisés dans la sécurité des ressources de processus
MQNLIST	GMQNLISTE	Profils utilisés dans la sécurité des ressources de liste de noms
MXNLIST	GMXNLIST	Profils en casse mixte et en majuscules utilisés dans la sécurité des ressources de liste de noms
MXTOPIC	GMXTOPIC	Profils en casse mixte et en majuscules utilisés dans la sécurité des rubriques

Certaines classes possèdent une *classe de groupe* associée qui vous permet de regrouper des groupes de ressources ayant des exigences d'accès similaires. Pour plus de détails sur la différence entre les classes de membre et de groupe et sur la date d'utilisation d'une classe de membre ou de groupe, voir le manuel z/OS Security Server RACF Security Administrator's Guide.

Les classes doivent être activées pour que des contrôles de sécurité puissent être effectués. Pour activer toutes les classes IBM MQ , vous pouvez utiliser la commande RACF suivante:

```
SETROPTS CLASSACT(MQADMIN,MXADMIN,MQQUEUE,MXQUEUE,MQPROC,MXPROC,
                  MQNLIST,MXNLIST,MXTOPIC,MQCONN,MQCMDS)
```

Vous devez également vous assurer que vous avez configuré les classes pour qu'elles puissent accepter les profils génériques. Vous pouvez également effectuer cette opération à l'aide de la commande RACF SETROPTS, par exemple:

```
SETROPTS GENERIC(MQADMIN,MXADMIN,MQQUEUE,MXQUEUE,MQPROC,MXPROC,
                 MQNLIST,MXNLIST,MXTOPIC,MQCONN,MQCMDS)
```

z/OS RACF profils

Tous les profils RACF utilisés par IBM MQ contiennent un préfixe, qui correspond au nom du gestionnaire de files d'attente ou au nom du groupe de partage de files d'attente. Soyez prudent lorsque vous utilisez le signe pourcentage comme caractère générique.

Tous les profils RACF utilisés par IBM MQ contiennent un préfixe. Pour la sécurité au niveau du groupe de partage de files d'attente, il s'agit du nom du groupe de partage de files d'attente. Pour la sécurité au niveau du gestionnaire de files d'attente, le préfixe est le nom du gestionnaire de files d'attente. Si vous utilisez un mélange de sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, vous utiliserez des profils avec les deux types de préfixe. (La sécurité au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente est décrite dans ConceptsIBM MQ for z/OS : sécurité.)

Par exemple, si vous souhaitez protéger une file d'attente appelée QUEUE_FOR_SUBSCRIBER_LIST dans le groupe de partage de files d'attente QSG1 au niveau du groupe de partage de files d'attente, le profil approprié sera défini dans RACF comme suit:

```
RDEFINE MQQUEUE QSG1.QUEUE_FOR_SUBSCRIBER_LIST
```

Si vous souhaitez protéger une file d'attente appelée QUEUE_FOR_LOST_CARD_LIST, qui appartient au gestionnaire de files d'attente STCD au niveau du gestionnaire de files d'attente, le profil approprié est défini sur RACF comme suit:

```
RDEFINE MQQUEUE STCD.QUEUE_FOR_LOST_CARD_LIST
```

Cela signifie que différents gestionnaires de files d'attente et groupes de partage de files d'attente peuvent partager la même base de données RACF tout en ayant des options de sécurité différentes.

N'utilisez pas de noms de gestionnaire de files d'attente génériques dans les profils pour éviter un accès utilisateur imprévu.

IBM MQ permet d'utiliser le signe de pourcentage (%) dans les noms d'objet. Toutefois, RACF utilise le caractère% comme caractère générique à caractère unique. Cela signifie que lorsque vous définissez un nom d'objet avec un caractère% dans son nom, vous devez le prendre en compte lorsque vous définissez le profil correspondant.

Par exemple, pour la file d'attente CREDIT_CARD_ %_RATE_INQUIRY, sur le gestionnaire de files d'attente CRDP, le profil est défini sur RACF comme suit:

```
RDEFINE MQQUEUE CRDP.CREDIT_CARD_%_RATE_INQUIRY
```

Cette file d'attente ne peut pas être protégée par un profil générique, tel que CRDP. * *.

IBM MQ permet d'utiliser des majuscules et des minuscules dans les noms d'objet. Vous pouvez protéger ces objets en définissant:

1. Des profils à casse mixte dans les classes RACF à casse mixte appropriées, ou
2. Profils génériques dans les classes RACF en majuscules appropriées.

Pour utiliser des profils de casse mixte et des classes RACF à casse mixte, vous devez suivre les étapes décrites dans «z/OS Migration d'un gestionnaire de files d'attente vers la sécurité à casse mixte», à la page 276.

Certains profils, ou parties de profils, ne restent en majuscules que lorsque les valeurs sont fournies par IBM MQ. Il s'agit des fonctions suivantes :

- Profils de commutation.
- Tous les qualificatifs de haut niveau (HLQ), y compris les identificateurs de sous-système et de groupe de partage de files d'attente.
- Profils des objets SYSTEM.
- Profils pour les objets par défaut.
- La classe **MQCMDS** , de sorte que tous les profils de commande sont en majuscules uniquement.
- La classe **MQCONN** , de sorte que tous les profils de connexion soient en majuscules uniquement.
- Profils **RESLEVEL** .
- La qualification `'object'` dans les profils de ressource de commande ; par exemple, `hlq.QUEUE.queuename`. Le nom de la ressource est uniquement composé de majuscules et de minuscules.
- Profils de file d'attente dynamique `hlq.CSQOREXX.*`, `hlq.CSQUTIL.*`et `CSQXCMD.*`.
- Le `'CONTEXT'` partie de `hlq.CONTEXT.resourcename`.
- La partie `'ALTERNATE.USER'` de `hlq.ALTERNATE.USER.userid`.

Par exemple, si vous avez une file d'attente appelée `PAYROLL.Dept1` sur le gestionnaire de files d'attente `QM01` et que vous utilisez:

- Profils à casse mixte ; vous pouvez définir un profil dans la IBM MQ classe RACF `MXQUEUE`

```
RDEFINE MXQUEUE MQ01.PAYROLL.Dept1
```

- Profils en majuscules ; vous pouvez définir un profil dans la IBM MQ classe RACF `MQQUEUE`

```
RDEFINE MQQUEUE MQ01.PAYROLL.*
```

Le premier exemple, qui utilise des profils à casse mixte, vous donne un contrôle plus granulaire sur l'octroi des droits d'accès à la ressource.

z/OS Profils de commutation

Pour contrôler la vérification de la sécurité effectuée par IBM MQ, vous utilisez des *profils de commutation*. Un profil de commutation est un profil RACF normal qui a une signification spéciale pour IBM MQ. La liste d'accès dans les profils de commutateur n'est pas utilisée par IBM MQ.

IBM MQ gère un commutateur interne pour chaque type de commutateur affiché dans les tableaux Profils de commutation pour la sécurité au niveau du sous-système, Profils de commutation pour la sécurité au niveau du groupe de partage de files d'attente ou du gestionnaire de files d'attenteet Profils de commutation pour la vérification des ressources. Les profils de commutation peuvent être gérés au niveau du groupe de partage de files d'attente, au niveau du gestionnaire de files d'attente ou au niveau d'une combinaison des deux. A l'aide d'un ensemble unique de profils de commutation de sécurité de groupe de partage de files d'attente, vous pouvez contrôler la sécurité sur tous les gestionnaires de files d'attente au sein d'un groupe de partage de files d'attente.

Lorsqu'un commutateur de sécurité est défini, les contrôles de sécurité associés au commutateur sont effectués. Lorsqu'un commutateur de sécurité est désactivé, les contrôles de sécurité associés au commutateur sont ignorés. Par défaut, tous les commutateurs de sécurité sont activés.

z/OS *Commutateurs et classes*

Lorsque vous démarrez un gestionnaire de files d'attente ou que vous actualisez la sécurité, IBM MQ définit les commutateurs en fonction de l'état des différentes classes RACF .

Lorsqu'un gestionnaire de files d'attente est démarré (ou lorsque la classe MQADMIN ou MXADMIN est actualisée par la commande IBM MQ REFRESH SECURITY), IBM MQ vérifie d'abord le statut de RACF et la classe appropriée:

- La classe MQADMIN si vous utilisez des profils en majuscules
- La classe MXADMIN si vous utilisez un profil à casse mixte.

Il désactive l'option de sécurité du sous-système si l'une des conditions suivantes est vérifiée:

- RACF est inactif ou n'est pas installé.
- La classe MQADMIN ou MXADMIN n'est pas définie (ces classes sont toujours définies pour RACF car elles sont incluses dans la table des descripteurs de classe (CDT)).
- La classe MQADMIN ou MXADMIN n'a pas été activée.

Si RACF et la classe MQADMIN ou MXADMIN sont toutes deux actives, IBM MQ vérifie la classe MQADMIN ou MXADMIN pour voir si des profils de commutation ont été définis. Il vérifie d'abord les profils décrits dans «Profils pour le contrôle de la sécurité du sous-système», à la page 192. Si la sécurité du sous-système n'est pas requise, IBM MQ désactive l'option de sécurité du sous-système interne et n'effectue aucune autre vérification.

Les profils déterminent si le commutateur IBM MQ correspondant est défini sur ou hors fonction.

- Si le commutateur est désactivé, ce type de sécurité est désactivé.
- Si un commutateur IBM MQ est défini sur on, IBM MQ vérifie le statut de la classe RACF associée au type de sécurité correspondant au commutateur IBM MQ . Si la classe n'est pas installée ou n'est pas active, le commutateur IBM MQ est désactivé. Par exemple, les vérifications de sécurité de processus ne sont pas effectuées si la classe MQPROC ou MXPROC n'a pas été activée. La classe qui n'est pas active est équivalente à la définition de NO.PROCESS.CHECKS pour chaque gestionnaire de files d'attente et groupe de partage de files d'attente qui utilise cette base de données RACF .

z/OS *Fonctionnement des commutateurs*

Pour désactiver un commutateur de sécurité, définissez un NO.* Changez de profil pour cela. Vous pouvez remplacer un NO.* défini au niveau du groupe de partage de files d'attente en définissant un fichier YES.* pour un gestionnaire de files d'attente.

Pour désactiver un commutateur de sécurité, vous devez définir un NO.* Changez de profil pour cela. L'existence d'un NO.* profile signifie que les contrôles de sécurité ne sont **pas** effectués pour ce type de ressource, sauf si vous choisissez de remplacer un paramètre de niveau de groupe de partage de files d'attente sur un gestionnaire de files d'attente particulier. Ceci est décrit dans «Remplacement des paramètres de niveau de groupe de partage de files d'attente», à la page 192.

Si votre gestionnaire de files d'attente n'est pas membre d'un groupe de partage de files d'attente, vous n'avez pas besoin de définir de profils de niveau de groupe de partage de files d'attente ni de profils de substitution. Toutefois, n'oubliez pas de définir ces profils si le gestionnaire de files d'attente rejoint un groupe de partage de files d'attente à une date ultérieure.

Chaque NO.* Un profil de commutateur détecté par IBM MQ désactive la vérification de ce type de ressource. Les profils de commutation sont activés lors du démarrage du gestionnaire de files d'attente. Si vous modifiez les profils de commutation alors que des gestionnaires de files d'attente affectés sont en cours d'exécution, vous pouvez demander à IBM MQ de reconnaître les modifications en émettant la commande IBM MQ REFRESH SECURITY.

Les profils de commutateur doivent toujours être définis dans la classe MQADMIN ou MXADMIN. Ne les définissez pas dans la classe GMQADMIN ou GMXADMIN. Les tableaux Profils de commutation pour la sécurité au niveau du sous-système et Profils de commutation pour la vérification des ressources affichent les profils de commutation valides et le type de sécurité qu'ils contrôlent.

Remplacement des paramètres de niveau de groupe de partage de files d'attente

Vous pouvez redéfinir les paramètres de sécurité au niveau du groupe de partage de files d'attente pour un gestionnaire de files d'attente particulier membre de ce groupe. Si vous souhaitez effectuer des vérifications de gestionnaire de files d'attente sur un gestionnaire de files d'attente individuel qui ne sont pas effectuées sur d'autres gestionnaires de files d'attente du groupe, utilisez (qmgr-name.YES. *) profils de commutation.

A l'inverse, si vous ne souhaitez pas effectuer une vérification spécifique sur un gestionnaire de files d'attente particulier au sein d'un groupe de partage de files d'attente, définissez un (qmgr-name.NO. *) pour ce type de ressource particulier sur le gestionnaire de files d'attente et ne définissez pas de profil pour le groupe de partage de files d'attente. (IBM MQ ne recherche un profil de niveau groupe de partage de files d'attente que s'il ne trouve pas de profil de niveau gestionnaire de files d'attente.)

z/OS *Profils pour le contrôle de la sécurité du sous-système*

IBM MQ vérifie si des contrôles de sécurité du sous-système sont requis pour le sous-système, pour le gestionnaire de files d'attente et pour le groupe de partage de files d'attente.

Le premier contrôle de sécurité effectué par IBM MQ permet de déterminer si des contrôles de sécurité sont requis pour l'ensemble du sous-système IBM MQ . Si vous indiquez que vous ne voulez pas de sécurité de sous-système, aucune vérification supplémentaire n'est effectuée.

Les profils de commutateur suivants sont vérifiés pour déterminer si la sécurité du sous-système est requise. Figure 14, à la page 193 affiche l'ordre dans lequel ils sont vérifiés.

Tableau 24. Profils de commutation pour la sécurité au niveau du sous-système

Nom du profil de commutation	Type de ressource ou de vérification contrôlée
qmgr-name.NO.SUBSYS.SECURITY	Sécurité du sous-système pour ce gestionnaire de files d'attente
qsg-name.NO.SUBSYS.SECURITY	Sécurité du sous-système pour ce groupe de partage de files d'attente
qmgr-name.YES.SUBSYS.SECURITY	Remplacement de la sécurité du sous-système pour ce gestionnaire de files d'attente

Si votre gestionnaire de files d'attente n'est pas membre d'un groupe de partage de files d'attente, IBM MQ recherche uniquement le profil de commutation qmgr-name.NO.SUBSYS.SECURITY .

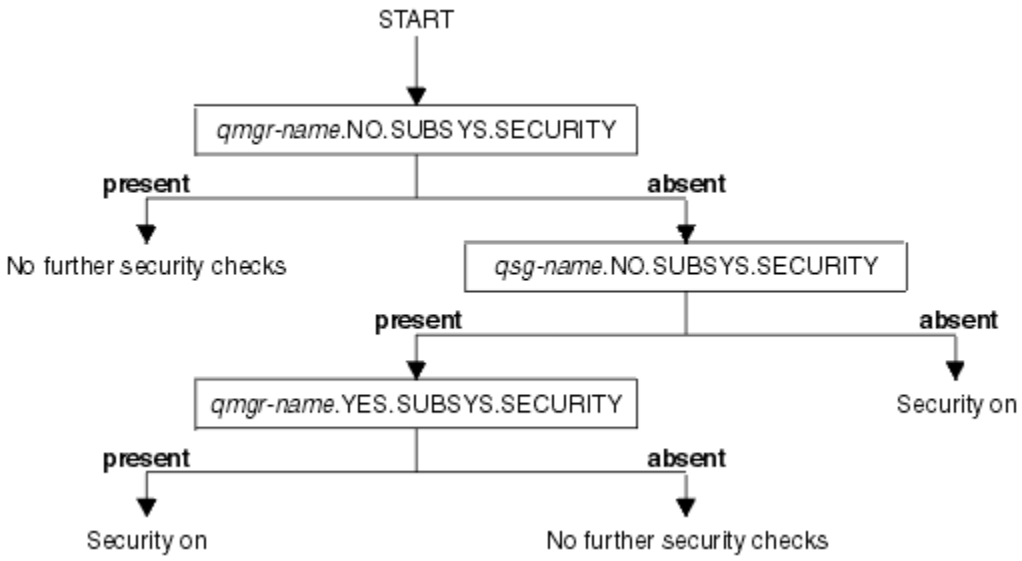


Figure 14. Vérification de la sécurité du sous-système

z/OS *Profils permettant de contrôler la sécurité au niveau du groupe de partage de files d'attente ou du gestionnaire de files d'attente*

Si une vérification de la sécurité du sous-système est requise, IBM MQ vérifie si une vérification de la sécurité est requise au niveau du groupe de partage de files d'attente ou du gestionnaire de files d'attente.

Lorsque IBM MQ a déterminé que la vérification de la sécurité est requise, il détermine ensuite si la vérification est requise au niveau du groupe de partage de files d'attente et/ou du gestionnaire de files d'attente. Ces vérifications ne sont pas effectuées si votre gestionnaire de files d'attente n'est pas membre d'un groupe de partage de files d'attente.

Les profils de commutateur suivants sont vérifiés pour déterminer le niveau requis. Figure 15, à la page 194 et Figure 16, à la page 194 affichent l'ordre dans lequel ils sont vérifiés.

Tableau 25. Profils de commutation pour la sécurité au niveau du groupe de partage de files d'attente ou du gestionnaire de files d'attente

Nom du profil de commutation	Type de ressource ou de vérification contrôlée
qmgr-name.NO.QMGR.CHECKS	Aucun contrôle au niveau du gestionnaire de files d'attente pour ce gestionnaire de files d'attente
qsg-name.NO.QMGR.CHECKS	Aucun contrôle au niveau du gestionnaire de files d'attente pour ce groupe de partage de files d'attente
qmgr-name.YES.QMGR.CHECKS	Remplacement des vérifications au niveau du gestionnaire de files d'attente pour ce gestionnaire de files d'attente
qmgr-name.NO.QSG.CHECKS	Aucun contrôle de niveau de groupe de partage de files d'attente pour ce gestionnaire de files d'attente
qsg-name.NO.QSG.CHECKS	Aucun contrôle de niveau de groupe de partage de files d'attente pour ce groupe de partage de files d'attente
qmgr-name.YES.QSG.CHECKS	Remplacement des vérifications de niveau groupe de partage de files d'attente pour ce gestionnaire de files d'attente

Si la sécurité du sous-système est active, vous ne pouvez pas désactiver la sécurité au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente. Si vous essayez de le faire, IBM MQ définit le contrôle de sécurité sur les deux niveaux.

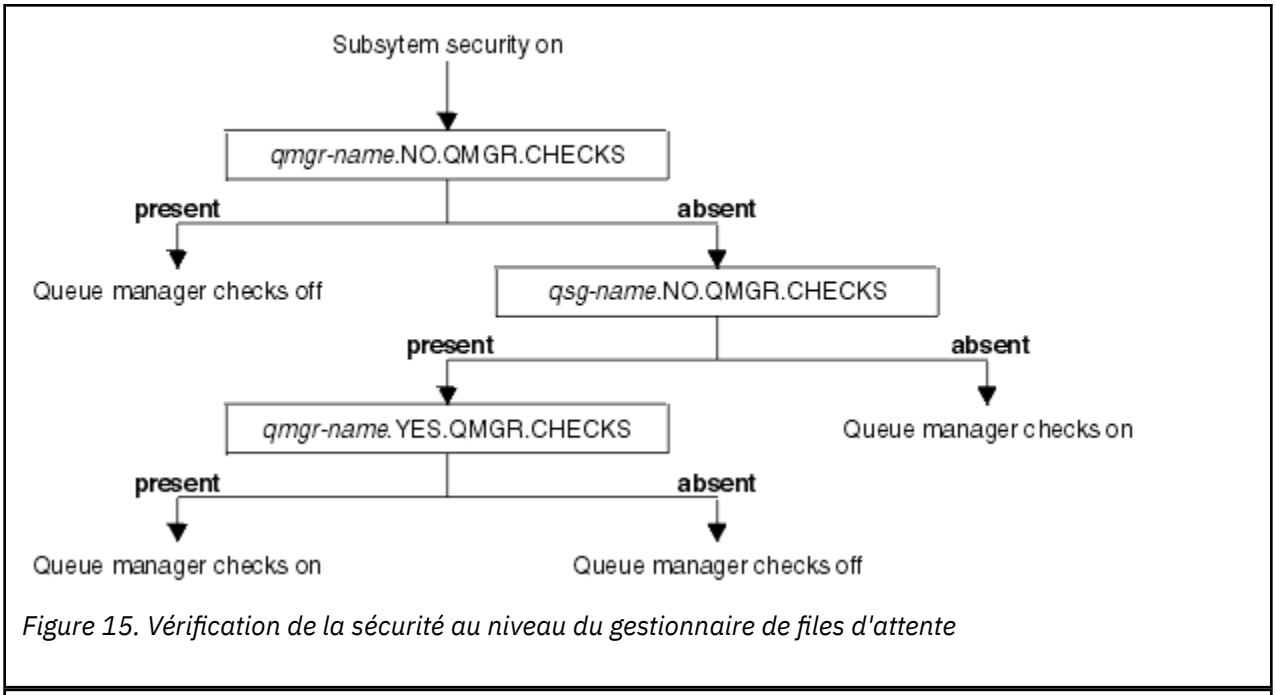


Figure 15. Vérification de la sécurité au niveau du gestionnaire de files d'attente

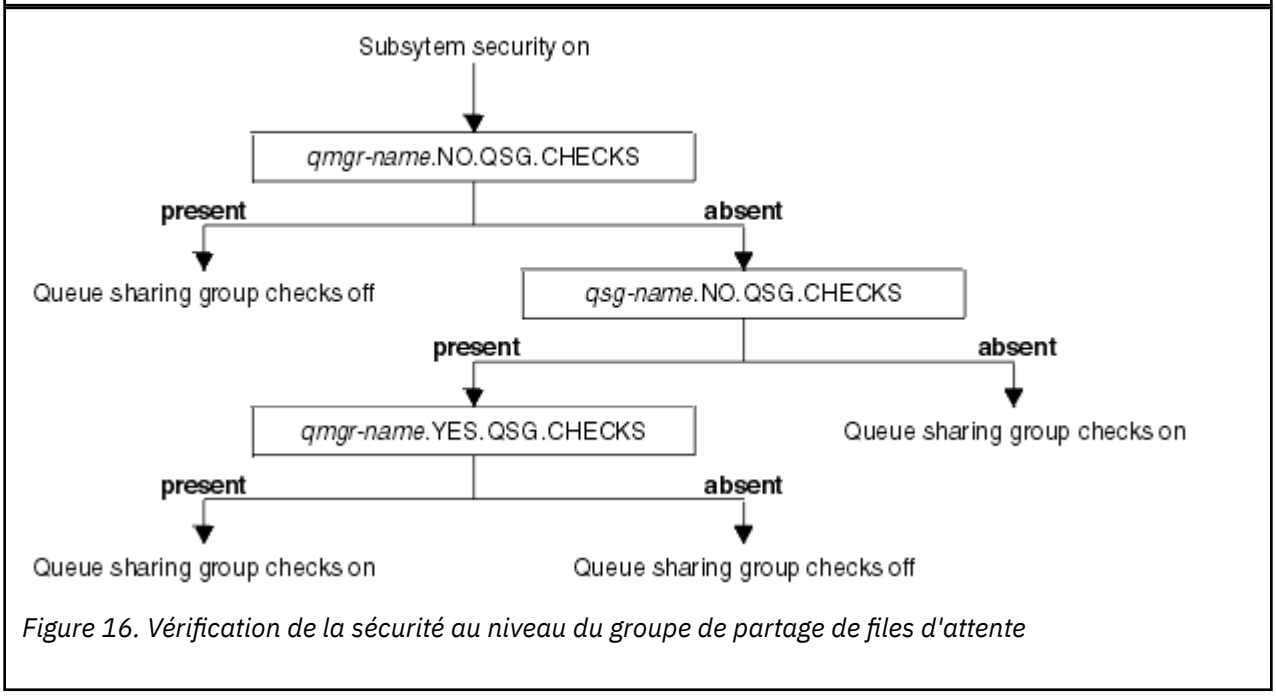


Figure 16. Vérification de la sécurité au niveau du groupe de partage de files d'attente

z/OS *Combinaisons valides de commutateurs de sécurité*

Seules certaines combinaisons de commutateurs sont valides. Si vous utilisez une combinaison de paramètres de commutateur non valide, le message CSQH026I est émis et le contrôle de sécurité est défini au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente.

Tableau 26, à la page 195, Tableau 27, à la page 195, Tableau 28, à la page 195et Tableau 29, à la page 196 affichent les ensembles de combinaisons de paramètres de commutateur valides pour chaque type de niveau de sécurité.

Tableau 26. Combinaisons de commutateurs de sécurité valides pour la sécurité au niveau du gestionnaire de files d'attente

Combinaisons
qmgr-name.NO.QSG.CHECKS
qsg-name.NO.QSG.CHECKS
qmgr-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS
qsg-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS

Tableau 27. Combinaisons de commutateurs de sécurité valides pour la sécurité au niveau du groupe de partage de files d'attente

Combinaisons
qmgr-name.NO.QMGR.CHECKS
qsg-name.NO.QMGR.CHECKS
qmgr-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS

Tableau 28. Combinaisons de commutateurs de sécurité valides pour la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente

Combinaisons
qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS Aucun QSG.* profils définis
Pas de QMGR.* profils définis qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
Aucun profil défini pour l'un ou l'autre des commutateurs

*Tableau 29. Autres combinaisons de commutateurs de sécurité valides qui commutent les deux niveaux de vérification **on**.*
Combinaisons
qmgr-name.NO.QMGR.CHECKS qmgr-name.NO.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS
qmgr-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qmgr-name.NO.QSG.CHECKS

z/OS *Vérifications au niveau de la ressource*

Un certain nombre de profils de commutation sont utilisés pour contrôler l'accès aux ressources. Certaines vérifications d'arrêt sont effectuées sur un gestionnaire de files d'attente ou un groupe de partage de files d'attente. Ils peuvent être remplacés par des profils qui permettent de vérifier des gestionnaires de files d'attente spécifiques.

La Tableau 30, à la page 196 présente les profils de commutation utilisés pour contrôler l'accès aux ressources IBM MQ .

Si votre gestionnaire de files d'attente fait partie d'un groupe de partage de files d'attente et que la sécurité du gestionnaire de files d'attente et du groupe de partage de files d'attente est active, vous pouvez utiliser un YES.* changer de profil pour remplacer les profils de niveau de groupe de partage de files d'attente et activer spécifiquement la sécurité pour un gestionnaire de files d'attente particulier.

Certains profils s'appliquent à la fois aux gestionnaires de files d'attente et aux groupes de partage de files d'attente. Ils sont préfixés par la chaîne *hlq* et vous devez remplacer le nom de votre groupe de partage de files d'attente ou de votre gestionnaire de files d'attente, selon le cas. Les noms de profil affichés avec le préfixe *qmgr-name* sont des profils de substitution de gestionnaire de files d'attente ; vous devez remplacer le nom de votre gestionnaire de files d'attente.

Tableau 30. Profils de commutation pour la vérification des ressources

Type de vérification de ressource contrôlée	**Nom du profil de commutation**	**Profil de substitution pour un gestionnaire de files d'attente particulier**
Sécurité des connexions	hlq.NO.CONNECT.CHECKS	qmgr-name.YES.CONNECT.CHECKS
Sécurité de file d'attente	hlq.NO.QUEUE.CHECKS	qmgr-name.YES.QUEUE.CHECKS
Sécurité des processus	hlq.NO.PROCESS.CHECKS	qmgr-name.YES.PROCESS.CHECKS
Sécurité de la liste de noms	hlq.NO.NLIST.CHECKS	qmgr-name.YES.NLIST.CHECKS
sécurité du contexte	hlq.NO.CONTEXT.CHECKS	qmgr-name.YES.CONTEXT.CHECKS
sécurité de substitution	hlq.NO.ALTERNATE.USER.CHECKS	qmgr-name.YES.ALTERNATE.USER.CHECKS

Tableau 30. Profils de commutation pour la vérification des ressources (suite)

Type de vérification de ressource contrôlée	Nom du profil de commutation	Profil de substitution pour un gestionnaire de files d'attente particulier
Sécurité de commande	hlq.NO.CMD.CHECKS	qmgr-name.YES.CMD.CHECKS
Sécurité des ressources de commandes	hlq.NO.CMD.RESC.CHECKS	qmgr-name.YES.CMD.RESC.CHECKS
Sécurité des rubriques	hlq.NO.TOPIC.CHECKS	qmgr-name.YES.TOPIC.CHECKS
Remarque : Profils de commutation génériques tels que hlq.NO. * * sont ignorés par IBM MQ		

Par exemple, si vous souhaitez effectuer des vérifications de sécurité de processus sur le gestionnaire de files d'attente QM01, qui est membre du groupe de partage de files d'attente QSG3 mais que vous ne souhaitez pas effectuer de vérifications de sécurité de processus sur les autres gestionnaires de files d'attente du groupe, définissez les profils de commutation suivants:

```
QSG3.NO.PROCESS.CHECKS
QM01.YES.PROCESS.CHECKS
```

Si vous souhaitez que des contrôles de sécurité de file d'attente soient effectués sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente, à l'exception de QM02, définissez le profil de commutation suivant:

```
QM02.NO.QUEUE.CHECKS
```

(Il n'est pas nécessaire de définir un profil pour le groupe de partage de files d'attente car les vérifications sont automatiquement activées si aucun profil n'est défini.)

z/OS *Exemple de définition de commutateurs*

Les différents sous-systèmes IBM MQ ont des exigences de sécurité différentes, qui peuvent être implémentées à l'aide de profils de commutateur différents.

Quatre sous-systèmes IBM MQ ont été définis:

- MQP1 (système de production)
- MQP2 (système de production)
- MQD1 (système de développement)
- MQT1 (système de test)

Les quatre gestionnaires de files d'attente sont membres du groupe de partage de files d'attente QS01. Toutes les classes IBM MQ RACF ont été définies et activées.

Ces sous-systèmes ont des exigences de sécurité différentes:

- Les systèmes de production nécessitent un contrôle de sécurité IBM MQ complet pour être actifs au niveau du groupe de partage de files d'attente sur les deux systèmes.

  Pour ce faire, spécifiez le profil suivant:

  ```
  RDEFINE MQADMIN QS01.NO.QMGR.CHECKS
  ```

  Cette option permet de définir la vérification au niveau du groupe de partage de files d'attente pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Vous n'avez pas besoin de

définir d'autres profils de commutation pour les gestionnaires de files d'attente de production car vous souhaitez tout vérifier pour ces systèmes.

- Le gestionnaire de files d'attente de test MQT1 requiert également un contrôle de sécurité complet. Toutefois, comme vous souhaiterez peut-être le modifier ultérieurement, la sécurité peut être définie au niveau du gestionnaire de files d'attente de sorte que vous pouvez modifier les paramètres de sécurité de ce gestionnaire de files d'attente sans affecter les autres membres du groupe de partage de files d'attente.

  Pour ce faire, définissez la valeur NO.QSG.CHECKS pour MQT1 , comme suit:

  ```
  RDEFINE MQADMIN MQT1.NO.QSG.CHECKS
  ```

- Le gestionnaire de files d'attente de développement MQD1 a des exigences de sécurité différentes de celles du reste du groupe de partage de files d'attente. Elle requiert uniquement l'activation de la sécurité de la connexion et de la file d'attente.

  Pour cela, vous devez définir un profil `MQD1.YES.QMGR.CHECKS` pour ce gestionnaire de files d'attente, puis définir les profils suivants pour désactiver la vérification de la sécurité des ressources qui n'ont pas besoin d'être vérifiées:

  ```
  RDEFINE MQADMIN MQD1.NO.CMD.CHECKS
  RDEFINE MQADMIN MQD1.NO.CMD.RESC.CHECKS
  RDEFINE MQADMIN MQD1.NO.PROCESS.CHECKS
  RDEFINE MQADMIN MQD1.NO.NLIST.CHECKS
  RDEFINE MQADMIN MQD1.NO.CONTEXT.CHECKS
  RDEFINE MQADMIN MQD1.NO.ALTERNATE.USER.CHECKS
  ```

Lorsque le gestionnaire de files d'attente est actif, vous pouvez afficher les paramètres de sécurité en cours en émettant la commande DISPLAY SECURITY MQSC.

Vous pouvez également modifier les paramètres de commutation lorsque le gestionnaire de files d'attente est en cours d'exécution en définissant ou en supprimant le profil de commutation approprié dans la classe MQADMIN. Pour que les modifications apportées aux paramètres de commutateur soient actives, vous devez exécuter la commande REFRESH SECURITY pour la classe MQADMIN.

Voir «Régénération de la sécurité du gestionnaire de files d'attente sous z/OS», à la page 256 pour plus de détails sur l'utilisation des commandes DISPLAY SECURITY et REFRESH SECURITY.

z/OS Profils utilisés pour contrôler l'accès aux ressources IBM MQ

Vous devez définir des profils RACF pour contrôler l'accès aux ressources IBM MQ , en plus des profils de commutation qui peuvent avoir été définis. Cette collection de rubriques contient des informations sur les profils RACF pour les différents types de ressource IBM MQ .

Si aucun profil de ressource n'est défini pour un contrôle de sécurité particulier et qu'un utilisateur émet une demande impliquant ce contrôle, IBM MQ refuse l'accès. Il n'est pas nécessaire de définir des profils pour les types de sécurité associés aux commutateurs de sécurité que vous avez désactivés.

z/OS *Profils pour la sécurité de connexion*

Si la sécurité de connexion est active, vous devez définir des profils dans la classe MQCONN et autoriser les groupes ou les ID utilisateur nécessaires à accéder à ces profils, afin qu'ils puissent se connecter à IBM MQ.

Pour permettre l'établissement d'une connexion, vous devez accorder aux utilisateurs RACF un accès en lecture (READ) au profil approprié. (Si aucun profil de niveau gestionnaire de files d'attente n'existe et que votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente, des vérifications peuvent être effectuées sur les profils de niveau groupe de partage de files d'attente, si la sécurité est configurée pour cela.)

Un profil de connexion qualifié avec un nom de gestionnaire de files d'attente contrôle l'accès à un gestionnaire de files d'attente spécifique et les utilisateurs ayant accès à ce profil peuvent se connecter

à ce gestionnaire de files d'attente. Un profil de connexion qualifié avec le nom de groupe de partage de files d'attente contrôle l'accès à tous les gestionnaires de files d'attente du groupe de partage de files d'attente pour ce type de connexion. Par exemple, un utilisateur ayant accès à `QS01.BATCH` peut utiliser une connexion par lots à n'importe quel gestionnaire de files d'attente du groupe de partage de files d'attente QS01 qui n'a pas de profil de niveau gestionnaire de files d'attente défini.

Remarque :

1. Pour plus d'informations sur les ID utilisateur vérifiés pour différentes demandes de sécurité, voir «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244.
2. Des vérifications de sécurité au niveau des ressources (RESLEVEL) sont également effectuées au moment de la connexion. Pour plus de détails, voir «Profil de sécurité RESLEVEL», à la page 238.

La sécurité IBM MQ reconnaît les différents types de connexion suivants:

- Les connexions par lots (et de type lot) sont les suivantes:
  - z/OS travaux par lots
  - Applications TSO
  - Connexions USS
  - Db2Procédures mémorisées
- Connexions CICS
- Connexions IMS à partir des régions de contrôle et de traitement d'application
- Initiateur de canal IBM MQ

z/OS *Profils de sécurité de connexion pour les connexions par lots*

Les profils de vérification des connexions de type lot sont composés du nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente suivi du mot *BATCH*. Accordez à l'ID utilisateur associé à l'espace adresse de connexion l'accès en lecture (READ) au profil de connexion.

Les profils pour la vérification des connexions par lots et par lots prennent la forme suivante:

```
hlq.BATCH
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente). Si vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente. S'il ne parvient pas à trouver l'un des profils, la demande de connexion échoue.

Pour les demandes de connexion par lots ou par lots, vous devez autoriser l'ID utilisateur associé à l'espace adresse de connexion à accéder au profil de connexion. Par exemple, la commande RACF suivante permet aux utilisateurs du groupe CONNTQM1 de se connecter au gestionnaire de files d'attente TQM1; ces ID utilisateur seront autorisés à utiliser n'importe quelle connexion de type lot ou lot.

```
RDEFINE MQCONN TQM1.BATCH UACC(NONE)
PERMIT TQM1.BATCH CLASS(MQCONN) ID(CONNTQM1) ACCESS(READ)
```

z/OS *Utilisation de **CHCKLOCL** sur des applications liées localement*

CHCKLOCL s'applique uniquement aux connexions établies via des connexions BATCH et ne s'applique pas aux connexions établies à partir de CICS ou IMS. Les connexions établies via l'initiateur de canal sont contrôlées par **CHCKCLNT**.

Présentation

Si vous souhaitez configurer votre gestionnaire de files d'attente z/OS pour qu'il demande la vérification de l'ID utilisateur et du mot de passe pour certaines, mais pas pour toutes, de vos applications liées localement, vous devez effectuer une configuration supplémentaire.

En effet, une fois que **CHCKLOCL** (*REQUIRED*) est configuré, les applications par lots existantes qui utilisent l'appel API MQCONN ne peuvent plus se connecter au gestionnaire de files d'attente.

Pour z/OS uniquement, un mécanisme plus granulaire basé sur la sécurité de connexion d'un espace adresse peut être utilisé pour rétromigrer la configuration globale CHCKLOCL (REQUIRED) vers CHCKLOCL (OPTIONAL) pour les ID utilisateur spécifiquement définis. Le mécanisme utilisé est décrit dans le texte qui suit, ainsi qu'un exemple.

Afin d'autoriser plus de granularité sur **CHCKLOCL** (*REQUIRED*) que tout le monde, vous modifiez **CHCKLOCL** de la même manière que vous modifiez le niveau d'accès de l'ID utilisateur associé à l'espace adresse de connexion aux profils de connexion `hlq.batch` dans la classe MQCONN.

Si l'ID utilisateur de l'espace adresse ne dispose que d'un accès en lecture (READ), ce qui est le minimum requis pour pouvoir se connecter, la configuration **CHCKLOCL** s'applique comme indiqué.

Si l'ID utilisateur de l'espace adresse dispose d'un accès UPDATE (ou supérieur), la configuration **CHCKLOCL** fonctionne en mode *OPTIONAL* . En d'autres mots, vous n'avez pas besoin de fournir un ID utilisateur et un mot de passe, mais dans ce cas, l'ID utilisateur et le mot de passe doivent être une paire valide.

La sécurité de connexion est déjà configurée pour votre gestionnaire de files d'attente z/OS

Si la sécurité de connexion est configurée pour votre gestionnaire de files d'attente z/OS et que vous souhaitez que **CHCKLOCL** (*REQUIRED*) s'applique aux applications WAS liées localement, et à aucune autre application, procédez comme suit:

1. Commencez par **CHCKLOCL** (*FACULTATIF*) comme configuration. Cela signifie que la validité des ID utilisateur et des mots de passe fournis est vérifiée, mais qu'ils ne sont pas obligatoires.
2. Répertoriez tous les utilisateurs qui ont accès aux profils de sécurité de connexion en exécutant la commande suivante:

   ```
   RLIST MQCONN MQ23.BATCH AUTHUSER
   ```

   Cette commande affiche, par exemple:

   ```
   CLASS    NAME
   -----    ----
   MQCONN   MQ23.BATCH

   USER     ACCESS  ACCESS COUNT
   ----     ------  ------ -----
   JOHNDOE  READ    000009
   JDOE1    READ    000003
   WASUSER  READ    000000
   ```

3. Pour chaque ID utilisateur répertorié comme disposant d'un accès en lecture (READ), modifiez l'accès à

   ```
   UPDATE:- PERMIT MQ23.BATCH CLASS(MQCONN) ID(JOHNDOE) ACCESS(UPDATE)
   ```

4. Mettez à jour la configuration IBM MQ sur **CHCKLOCL** (*REQUIRED*).

   La combinaison de l'accès UPDATE à `MQ23.BATCH` et du paramètre en cours signifie que vous utilisez **CHCKLOCL** (*FACULTATIF*).
5. A présent, appliquez le comportement **CHCKLOCL** (*REQUIRED*) à un ID utilisateur spécifique, par exemple `WASUSER`, de sorte que toutes les connexions provenant de cette région doivent fournir un ID utilisateur et un mot de passe.

Pour ce faire, annulez la modification que vous avez apportée précédemment en exécutant la commande suivante:

```
PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
```

La sécurité de connexion n'est pas configurée pour votre gestionnaire de files d'attente z/OS

Dans cette situation, vous devez:

1. Créez des profils de connexion pour `hlq.`BATCH dans la classe MQCONN en exécutant la commande suivante:

   ```
   RDEFINE MQCONN MQ23.BATCH UACC(NONE)
   ```

2. Autorisez tous les ID utilisateur qui créent des connexions par lots au gestionnaire de files d'attente, de sorte qu'ils disposent d'un accès en mise à jour à ce profil. Cette opération ignore les exigences **CHCKLOCL** (*REQUIRED*) pour l'ID utilisateur et le mot de passe au moment de la connexion.

   Pour ce faire, exécutez la commande suivante:

   ```
   PERMIT MQ23.BATCH CLASS(MQCONN)ID(JOHNDOE) ACCESS(UPDATE)
   ```

   Il s'agit notamment des ID utilisateur:

   a. Utilisé pour les CSQUTIL, les panneaux ISPF et les autres outils liés localement.

   b. Associé à des connexions par lots au gestionnaire de files d'attente. Prenons l'exemple des procédures stockées Advanced Message Security, IBM Integration Bus, Db2 , des utilisateurs USS et TSO et des applications Java .

3. Supprimez le profil de commutation du gestionnaire de files d'attente en exécutant la commande suivante:

   ```
   hlq.NO.CONNECT.CHECKS
   ```

4. A présent, appliquez le comportement **CHCKLOCL** (*REQUIRED*) à un ID utilisateur spécifique, par exemple WASUSER, de sorte que toutes les connexions provenant de cette région doivent fournir un ID utilisateur et un mot de passe.

   Pour ce faire, annulez la modification que vous avez apportée précédemment en exécutant la commande suivante:

   ```
   PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
   ```

z/OS *Profils de sécurité de connexion pour les connexions CICS*

Les profils de vérification des connexions CICS sont composés du nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente suivi du mot *CICS* . Accordez à l'ID utilisateur associé à l'espace adresse CICS un accès en lecture (READ) au profil de connexion.

Les profils de vérification des connexions à partir de CICS se présentent sous la forme suivante:

```
hlq.CICS
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente). Si vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche un profil préfixé par le nom du gestionnaire de

files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente. S'il ne parvient pas à trouver l'un des profils, la demande de connexion échoue

Pour les demandes de connexion par CICS, vous devez uniquement autoriser l'ID utilisateur de l'espace adresse CICS à accéder au profil de connexion.

Par exemple, les commandes RACF suivantes permettent à l'ID utilisateur d'espace adresse CICS KCBCICS de se connecter au gestionnaire de files d'attente TQM1:

```
RDEFINE MQCONN TQM1.CICS UACC(NONE)
PERMIT TQM1.CICS CLASS(MQCONN) ID(KCBCICS) ACCESS(READ)
```

z/OS *Profils de sécurité de connexion pour les connexions IMS*

Les profils de vérification des connexions IMS sont composés du nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente suivi du mot *IMS* . Accordez à l'ID utilisateur de région dépendante et de contrôle IMS un accès en lecture (READ) au profil de connexion.

Les profils de vérification des connexions à partir de IMS se présentent sous la forme suivante:

```
hlq.IMS
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente). Si vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente. S'il ne parvient pas à trouver l'un des profils, la demande de connexion échoue

Pour les demandes de connexion par IMS, autorisez l'accès au profil de connexion pour les ID utilisateur de région dépendante et de contrôle IMS .

Par exemple, les commandes RACF suivantes permettent:

- ID utilisateur de la région IMS , IMSREG, permettant de se connecter au gestionnaire de files d'attente TQM1.
- Utilisateurs du groupe BMPGRP pour soumettre des travaux BMP.

```
RDEFINE MQCONN TQM1.IMS UACC(NONE)
PERMIT TQM1.IMS CLASS(MQCONN) ID(IMSREG,BMPGRP) ACCESS(READ)
```

z/OS *Profils de sécurité de connexion pour l'initiateur de canal*

Les profils de vérification des connexions à partir de l'initiateur de canal sont composés du nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente suivi du mot *CHIN*. Accordez à l'ID utilisateur utilisé par l'espace adresse de la tâche démarrée de l'initiateur de canal l'accès en lecture (READ) au profil de connexion.

Les profils permettant de vérifier les connexions à partir de l'initiateur de canal se présentent sous la forme suivante:

```
hlq.CHIN
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente). Si vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du

groupe de partage de files d'attente, IBM MQ recherche un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente. S'il ne parvient pas à trouver l'un des profils, la demande de connexion échoue

Pour les demandes de connexion par l'initiateur de canal, définissez l'accès au profil de connexion pour l'ID utilisateur utilisé par l'espace adresse de la tâche démarrée de l'initiateur de canal.

Par exemple, les commandes RACF suivantes permettent à l'espace adresse de l'initiateur de canal s'exécutant avec l'ID utilisateur DQCTRL de se connecter au gestionnaire de files d'attente TQM1:

```
RDEFINE MQCONN TQM1.CHIN UACC(NONE)
PERMIT TQM1.CHIN CLASS(MQCONN) ID(DQCTRL) ACCESS(READ)
```

z/OS *Profils pour la sécurité de la file d'attente*

Si la sécurité de file d'attente est active, vous devez définir des profils dans les classes appropriées et autoriser les groupes ou les ID utilisateur nécessaires à accéder à ces profils. Les profils de sécurité de file d'attente sont nommés d'après le gestionnaire de files d'attente ou le groupe de partage de files d'attente et la file d'attente à ouvrir.

Si la sécurité de la file d'attente est active, vous devez:

- Définissez des profils dans les classes **MQQUEUE** ou **GMQQUEUE** si vous utilisez des profils en majuscules.
- Définissez des profils dans les classes **MXQUEUE** ou **GMXQUEUE** si vous utilisez des profils à casse mixte.
- Autorisez les groupes ou les ID utilisateur nécessaires à accéder à ces profils, afin qu'ils puissent émettre des demandes d'API IBM MQ qui utilisent des files d'attente.

Les profils de sécurité de file d'attente se présentent sous la forme suivante:

```
hlq.queuename
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente) et `queuename` est le nom de la file d'attente en cours d'ouverture, comme indiqué dans le descripteur d'objet sur l'appel `MQOPEN` ou `MQPUT1` .

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'accès à une file d'attente unique sur ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'accès à une ou plusieurs files d'attente avec ce nom de file d'attente sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente, ou l'accès à une file d'attente partagée par n'importe quel gestionnaire de files d'attente du groupe. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau de gestionnaire de files d'attente pour cette file d'attente sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Si vous utilisez des files d'attente partagées, il est recommandé d'utiliser la sécurité au niveau du groupe de partage de files d'attente.

Pour plus de détails sur le fonctionnement de la sécurité de file d'attente lorsque le nom de file d'attente est celui d'un alias ou d'une file d'attente modèle z/OS, voir «Remarques relatives aux files d'attente alias», à la page 205 et «Remarques relatives aux files d'attente modèles», à la page 206 .

L'accès RACF requis pour ouvrir une file d'attente dépend des options `MQOPEN` ou `MQPUT1` indiquées. Si plusieurs des options MQOO_ * et MQPMO_ * sont codées, la vérification de la sécurité de la file d'attente est effectuée pour les droits RACF les plus élevés requis.

Tableau 31. Niveaux d'accès pour la sécurité de la file d'attente à l'aide des appels MQOPEN ou MQPUT1

Option MQOPEN ou MQPUT1	**RACF niveau d'accès requis pour hlq.queuename**
MQOO_BROWSE	READ
MQOO_INTERROGATION	READ
MQOO_BIND_ *	UPDATE
MQOO_ENTRÉE_ *	UPDATE
MQOO_OUTPUT ou MQPUT1	UPDATE
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	UPDATE
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	UPDATE
MQOO_SAVE_ALL_CONTEXT	UPDATE
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	UPDATE
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	UPDATE
MQOO_SET	ALTER

Par exemple, sur le gestionnaire de files d'attente IBM MQ QM77, tous les ID utilisateur du groupe RACF PAYGRP doivent être autorisés à extraire des messages ou à les insérer dans toutes les files d'attente dont le nom commence par'PAY.'. Pour ce faire, utilisez les commandes RACF suivantes:

```
RDEFINE MQQUEUE QM77.PAY.** UACC(NONE)
PERMIT QM77.PAY.** CLASS(MQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

De plus, tous les ID utilisateur du groupe PAYGRP doivent avoir accès aux messages placés dans des files d'attente qui ne respectent pas la convention de dénomination PAY. Exemple :

```
REQUEST_QUEUE_FOR_PAYROLL
SALARY.INCREASE.SERVER
REPLIES.FROM.SALARY.MODEL
```

Pour ce faire, vous pouvez définir des profils pour ces files d'attente dans la classe GMQQUEUE et donner accès à cette classe comme suit:

```
RDEFINE GMQQUEUE PAYROLL.EXTRAS UACC(NONE)
        ADDMEM(QM77.REQUEST_QUEUE_FOR_PAYROLL,
               QM77.SALARY.INCREASE.SERVER,
               QM77.REPLIES.FROM.SALARY.MODEL)
PERMIT PAYROLL.EXTRAS CLASS(GMQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

Remarque :

1. Si le niveau d'accès RACF d'une application à un profil de sécurité de file d'attente est modifié, les modifications ne sont prises en compte que pour les nouveaux descripteurs d'objet obtenus (c'est-à-dire, les nouveaux MQOPEN) pour cette file d'attente. Ces descripteurs déjà existants au moment de

la modification conservent leur accès à la file d'attente. Si une application doit utiliser son niveau d'accès modifié à la file d'attente plutôt que son niveau d'accès existant, elle doit fermer et rouvrir la file d'attente pour chaque descripteur d'objet qui nécessite la modification.

2. Dans l'exemple, le nom de gestionnaire de files d'attente QM77 peut également être le nom d'un groupe de partage de files d'attente.

D'autres types de contrôle de sécurité peuvent également se produire lors de l'ouverture de la file d'attente en fonction des options d'ouverture spécifiées et des types de sécurité actifs. z/OS Voir aussi «Profils pour la sécurité de contexte», à la page 221 et «Profils pour la sécurité des utilisateurs de remplacement», à la page 219. Pour un tableau récapitulatif présentant les options d'ouverture et l'autorisation de sécurité requise lorsque la sécurité de la file d'attente, du contexte et de l'utilisateur de remplacement est active, voir Tableau 36, à la page 211.

Si vous utilisez la fonction de publication / abonnement, vous devez tenir compte des éléments suivants. Lorsqu'une demande MQSUB est traitée, un contrôle de sécurité est effectué pour s'assurer que l'ID utilisateur qui effectue la demande dispose des droits d'accès requis pour insérer des messages dans la file d'attente IBM MQ cible ainsi que des droits d'accès requis pour s'abonner à la rubrique IBM MQ .

Tableau 32. Niveaux d'accès pour la sécurité de la file d'attente à l'aide de l'appel MQSUB

MQSUB, option	**RACF niveau d'accès requis pour hlq.queuename**
MQSO_ALTER, MQSO_CREATE et MQSO_RESUME	UPDATE

Remarque :

1. `hlq.queuename` est la file d'attente de destination des publications. Lorsqu'il s'agit d'une file d'attente gérée, vous devez accéder à la file d'attente modèle appropriée à utiliser pour la file d'attente gérée et la file d'attente dynamique créée.
2. Vous pouvez utiliser une telle technique pour la file d'attente de destination que vous fournissez sur un appel d'API MQSUB si vous souhaitez faire la distinction entre les utilisateurs qui effectuent les abonnements et les utilisateurs qui extraient les publications de la file d'attente de destination.

z/OS *Remarques relatives aux files d'attente alias*

Lorsque vous émettez un appel MQOPEN ou MQPUT1 pour une file d'attente alias, IBM MQ effectue une vérification des ressources par rapport au nom de file d'attente indiqué dans le descripteur d'objet (MQOD) de l'appel. Elle ne vérifie pas si l'utilisateur est autorisé à accéder au nom de la file d'attente cible.

Par exemple, une file d'attente alias appelée PAYROLL.REQUEST se résout en une file d'attente cible de PAY.REQUEST. Si la sécurité de la file d'attente est active, vous devez uniquement être autorisé à accéder à la file d'attente PAYROLL.REQUEST. Aucune vérification n'est effectuée pour déterminer si vous êtes autorisé à accéder à la file d'attente PAY.REQUEST.

z/OS *Utilisation de files d'attente alias pour faire la distinction entre les demandes MQGET et MQPUT*

La plage d'appels MQI disponibles dans un niveau d'accès peut entraîner un problème si vous souhaitez restreindre l'accès à une file d'attente pour n'autoriser que l'appel **MQPUT** ou l'appel **MQGET** . Une file d'attente peut être protégée en définissant deux alias qui se résolvent en cette file d'attente: un qui permet aux applications d'extraire des messages de la file d'attente et un qui permet aux applications d'insérer des messages dans la file d'attente.

Le texte suivant vous donne un exemple de la manière dont vous pouvez définir vos files d'attente dans IBM MQ:

```
DEFINE QLOCAL(MUST_USE_ALIAS_TO_ACCESS) GET(ENABLED)
       PUT(ENABLED)

DEFINE QALIAS(USE_THIS_ONE_FOR_GETS) GET(ENABLED)
```

```
       PUT(DISABLED) TARGET(MUST_USE_ALIAS_TO_ACCESS)

DEFINE QALIAS(USE_THIS_ONE_FOR_PUTS) GET(DISABLED)
       PUT(ENABLED) TARGET(MUST_USE_ALIAS_TO_ACCESS)
```

Vous devez également créer les définitions RACF suivantes:

```
RDEFINE MQQUEUE hlq.MUST_USE_ALIAS_TO_ACCESS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_GETS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_PUTS UACC(NONE)
```

Vérifiez ensuite qu'aucun utilisateur n'a accès à la file d'attente hlq.MUST_USE_ALIAS_TO_ACCESSSet accordez aux utilisateurs ou aux groupes appropriés l'accès à l'alias. Pour ce faire, utilisez les commandes RACF suivantes:

```
PERMIT hlq.USE_THIS_ONE_FOR_GETS CLASS(MQQUEUE)
       ID(GETUSER,GETGRP) ACCESS(UPDATE)
PERMIT hlq.USE_THIS_ONE_FOR_PUTS CLASS(MQQUEUE)
       ID(PUTUSER,PUTGRP) ACCESS(UPDATE)
```

Cela signifie que l'ID utilisateur GETUSER et les ID utilisateur du groupe GETGRP sont uniquement autorisés à obtenir des messages sur MUST_USE_ALIAS_TO_ACCESS via la file d'attente alias USE_THIS_ONE_FOR_GETS; et que l'ID utilisateur PUTUSER et les ID utilisateur du groupe PUTGRP sont uniquement autorisés à placer des messages via la file d'attente alias USE_THIS_ONE_FOR_PUTS.

Remarque :

1. Si vous souhaitez utiliser une telle technique, vous devez en informer vos développeurs d'applications afin qu'ils puissent concevoir leurs programmes de manière appropriée.
2. Vous pouvez utiliser une telle technique pour la file d'attente de destination que vous fournissez sur une demande d'API MQSUB si vous souhaitez faire la distinction entre les utilisateurs qui effectuent les abonnements et les utilisateurs qui "obtiennent" les publications de la file d'attente de destination.

z/OS *Remarques relatives aux files d'attente modèles*

Pour ouvrir une file d'attente modèle, vous devez pouvoir ouvrir à la fois la file d'attente modèle elle-même et la file d'attente dynamique dans laquelle elle est résolue. Définissez des profils RACF génériques pour les files d'attente dynamiques, y compris les files d'attente dynamiques utilisées par les utilitaires IBM MQ .

Lorsque vous ouvrez une file d'attente modèle, la sécurité IBM MQ effectue deux contrôles de sécurité de file d'attente:

1. Êtes-vous autorisé à accéder à la file d'attente modèle?
2. Êtes-vous autorisé à accéder à la file d'attente dynamique dans laquelle la file d'attente modèle est résolue?

Si le nom de la file d'attente dynamique contient un astérisque (*) de fin, ce caractère * est remplacé par une chaîne de caractères générée par IBM MQ, afin de créer une file d'attente dynamique avec un nom unique. Toutefois, comme le nom complet, y compris cette chaîne générée, est utilisé pour vérifier les droits d'accès, vous devez définir des profils génériques pour ces files d'attente.

Par exemple, un appel MQOPEN utilise le nom de file d'attente modèle CREDIT.CHECK.REPLY.MODEL et un nom de file d'attente dynamique CREDIT.REPLY.* sur le gestionnaire de files d'attente (ou le groupe de partage de files d'attente) MQSP.

Pour ce faire, vous devez exécuter les commandes RACF suivantes pour définir les profils de file d'attente nécessaires:

```
RDEFINE MQQUEUE MQSP.CREDIT.CHECK.REPLY.MODEL
RDEFINE MQQUEUE MQSP.CREDIT.REPLY.**
```

Vous devez également exécuter les commandes RACF PERMIT correspondantes pour autoriser l'utilisateur à accéder à ces profils.

Un nom de file d'attente dynamique typique créé par une commande MQOPEN est similaire à CREDIT.REPLY.A346EF00367849A0. La valeur précise du dernier qualificatif est imprévisible ; c'est pourquoi vous devez utiliser des profils génériques pour ces noms de file d'attente.

Un certain nombre d'utilitaires IBM MQ ont inséré des messages dans des files d'attente dynamiques. Vous devez définir des profils pour les noms de file d'attente dynamique suivants et fournir un accès RACF UPDATE aux ID utilisateur appropriés (voir «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244 pour les ID utilisateur corrects):

```
SYSTEM.CSQUTIL.*  (used by CSQUTIL)
SYSTEM.CSQOREXX.* (used by the operations and control panels)
SYSTEM.CSQXCMD.*  (used by the channel initiator when processing CSQINPX)
CSQ4SAMP.*        (used by the IBM MQ supplied samples)
```

Vous pouvez également envisager de définir un profil pour contrôler l'utilisation du nom de file d'attente dynamique utilisé par défaut dans les membres de copie de programme d'application. Les fichiers de stockage fournis par IBM MQcontiennent un *DynamicQName*par défaut, à savoir CSQ. *. Cela permet d'établir un profil RACF approprié.

Remarque : N'autorisez pas les programmeurs d'application à spécifier un seul * pour le nom de la file d'attente dynamique. Dans ce cas, vous devez définir un hlq. * * dans la classe MQQUEUE, et vous devez lui donner un accès étendu. Cela signifie que ce profil peut également être utilisé pour d'autres files d'attente non dynamiques qui n'ont pas de profil RACF plus spécifique. Vos utilisateurs peuvent donc accéder aux files d'attente auxquelles vous ne souhaitez pas qu'ils accèdent.

z/OS *Options de fermeture sur les files d'attente dynamiques permanentes*

Si une application ouvre une file d'attente dynamique permanente créée par une autre application, puis tente de supprimer cette file d'attente à l'aide de l'option MQCLOSE , des contrôles de sécurité supplémentaires sont appliqués lorsque la tentative est effectuée.

Tableau 33. Niveaux d'accès pour les options de fermeture sur les files d'attente dynamiques permanentes

MQCLOSE, option	**RACF niveau d'accès requis pour hlq.queuename**
MQCO_DELETE	ALTER
MQCO_DELETE_PURGE	ALTER

z/OS *Sécurité et files d'attente éloignées*

Lorsqu'un message est inséré dans une file d'attente éloignée, la sécurité de la file d'attente implémentée par le gestionnaire de files d'attente locales dépend de la manière dont la file d'attente éloignée est spécifiée lorsqu'elle est ouverte.

Les règles suivantes sont appliquées:

1. Si la file d'attente éloignée a été définie sur le gestionnaire de files d'attente local à l'aide de la commande IBM MQ DEFINE QREMOTE, la file d'attente vérifiée est le nom de la file d'attente éloignée. Par exemple, si une file d'attente éloignée est définie sur le gestionnaire de files d'attente MQS1 comme suit:

   ```
   DEFINE QREMOTE(BANK7.CREDIT.REFERENCE)
          RNAME(CREDIT.SCORING.REQUEST)
          RQMNAME(BNK7)
          XMITQ(BANK1.TO.BANK7)
   ```

   Dans ce cas, un profil pour BANK7.CREDIT.REFERENCE doit être définie dans la classe MQQUEUE.

2. Si le nom *ObjectQMgr* de la demande ne correspond pas au gestionnaire de files d'attente local, un contrôle de sécurité est effectué sur le nom du gestionnaire de files d'attente (distant) résolu, sauf dans le cas d'une file d'attente de cluster où le contrôle est effectué sur le nom de la file d'attente de cluster.

   Par exemple, la file d'attente de transmission BANK1.TO.BANK7 est défini sur le gestionnaire de files d'attente MQS1. Une demande MQPUT1 est ensuite émise sur MQS1 en spécifiant *ObjectName* comme BANK1.INTERBANK.TRANSFERS et *ObjectQMgrNom* de BANK1.TO.BANK7. Dans ce cas, l'utilisateur qui effectue la demande doit avoir accès à BANK1.TO.BANK7.
3. Si vous envoyez une demande MQPUT à une file d'attente et que vous spécifiez *ObjectQMgrName* comme nom d'alias du gestionnaire de files d'attente local, seul le nom de la file d'attente est vérifié pour la sécurité, et non celui du gestionnaire de files d'attente.

Lorsque le message est envoyé au gestionnaire de files d'attente éloignées, il peut faire l'objet d'un traitement de sécurité supplémentaire. Pour plus d'informations, voir «Sécurité de la messagerie distante», à la page 97.

z/OS *Sécurité de la file d'attente de rebut*

Des considérations particulières s'appliquent à la file d'attente des messages non livrés, car de nombreux utilisateurs doivent pouvoir y placer des messages, mais l'accès à l'extraction des messages doit être strictement limité. Pour ce faire, vous pouvez appliquer des droits RACF différents à la file d'attente de rebut et à une file d'attente alias.

Les messages non distribués peuvent être placés dans une file d'attente spéciale appelée file d'attente de rebut. Si vous disposez de données sensibles qui pourraient éventuellement se retrouver dans cette file d'attente, vous devez prendre en compte les implications de cette situation sur la sécurité car vous ne souhaitez pas que des utilisateurs non autorisés extraient ces données.

Chacun des éléments suivants doit être autorisé à insérer des messages dans la file d'attente de rebut:

- Programmes d'application.
- L'espace adresse de l'initiateur de canal et tous les ID utilisateur MCA. (Si le profil RESLEVEL n'est pas présent ou qu'il est défini de sorte que les ID utilisateur de canal soient vérifiés, l'ID utilisateur de canal doit également être autorisé à placer des messages dans la file d'attente de messages non livrés.)
- CKTI, l'initiateur de tâche CICS fourni par CICS.
- CSQQTRMN, le moniteur de déclenchement IMS fourni par IBM MQ.

La seule application qui peut extraire des messages de la file d'attente de messages non livrés doit être une application'spéciale'qui traite ces messages. Toutefois, un problème se produit si vous accordez aux applications le droit RACF UPDATE sur la file d'attente de rebut pour MQPUT s car elles peuvent alors extraire automatiquement des messages de la file d'attente à l'aide d'appels MQGET . Vous ne pouvez pas désactiver la file d'attente de rebut pour les opérations d'extraction car, si vous le faites, même les applications "spéciales" ne pourraient pas extraire les messages.

Une solution à ce problème consiste à configurer un accès à deux niveaux à la file d'attente des messages non livrés. CKTI, les transactions de l'agent de canal de message ou l'espace adresse de l'initiateur de canal, et les applications'spéciales'ont un accès direct ; les autres applications peuvent uniquement accéder à la file d'attente des messages non livrés via une file d'attente d'alias. Cet alias est défini pour permettre aux applications d'insérer des messages dans la file d'attente des messages non livrés, mais pas d'en extraire des messages.

Voici comment cela peut fonctionner:

1. Définissez la file d'attente de rebut réelle avec les attributs PUT (ENABLED) et GET (ENABLED), comme illustré dans l'exemple thlqual.SCSQPROC(CSQ4INYG).
2. Accordez à RACF le droit UPDATE pour la file d'attente de rebut aux ID utilisateur suivants:
   - ID utilisateur sous lesquels s'exécutent CKTI et les agents MCA ou l'espace adresse de l'initiateur de canal.
   - ID utilisateur associés à l'application de traitement de la file d'attente de rebut'spéciale'.

3. Définissez une file d'attente alias qui se résout en file d'attente de rebut réelle, mais attribuez à la file d'attente alias les attributs suivants: PUT (ENABLED) et GET (DISABLED). Attribuez à la file d'attente alias un nom ayant le même radical que le nom de la file d'attente de rebut, mais ajoutez les caractères ". PUT" à ce radical. Par exemple, si le nom de la file d'attente de rebut est hlq.DEAD.QUEUE, le nom de file d'attente alias serait hlq.DEAD.QUEUE.PUT.
4. Pour placer un message dans la file d'attente des messages non livrés, une application utilise la file d'attente alias. Voici ce que votre application doit faire:
   - Extrayez le nom de la vraie file d'attente de rebut. Pour ce faire, il ouvre l'objet gestionnaire de files d'attente à l'aide de `MQOPEN` , puis émet une commande `MQINQ` pour obtenir le nom de la file d'attente de rebut.
   - Générez le nom de la file d'attente alias en ajoutant les caractères'.PUT'à ce nom, dans ce cas, hlq.DEAD.QUEUE.PUT.
   - Ouvrez la file d'attente alias, hlq.DEAD.QUEUE.PUT.
   - Placez le message dans la file d'attente de rebut réelle en émettant une commande `MQPUT` sur la file d'attente alias.
5. Accordez à l'ID utilisateur associé à l'application le droit RACF UPDATE sur l'alias, mais pas d'accès (droit NONE) à la file d'attente de rebut réelle. Ce qui signifie que :
   - L'application peut placer des messages dans la file d'attente de rebut à l'aide de la file d'attente alias.
   - L'application ne peut pas extraire les messages de la file d'attente des messages non livrés à l'aide de la file d'attente alias car cette dernière est désactivée pour les opérations d'extraction.

   L'application ne peut pas extraire de messages de la file d'attente de rebut réelle car elle dispose des droits RACF appropriés.

Le Tableau 34, à la page 209 récapitule les droits d'accès RACF requis pour les différents participants à cette solution.

Tableau 34. Droits d'accès RACF à la file d'attente de rebut et à son alias

ID utilisateur associés	**File d'attente de rebut réelle (hlq.DEAD.QUEUE)**	**File d'attente de rebut d'alias (hlq.DEAD.QUEUE.PUT)**
Espace adresse MCA ou d'initiateur de canal et CKTI	UPDATE	AUCUN
Application "spéciale" (pour le traitement de la file d'attente de rebut)	UPDATE	AUCUN
ID utilisateur d'application écrits par l'utilisateur	AUCUN	UPDATE

Si vous utilisez cette méthode, l'application ne peut pas déterminer la longueur maximale des messages (MAXMSGL) de la file d'attente de rebut. En effet, l'attribut MAXMSGL ne peut pas être extrait d'une file d'attente alias. Par conséquent, votre application doit supposer que la longueur maximale des messages est de 100 Mo, la taille maximale prise en charge par IBM MQ for z/OS . La file d'attente de rebut réelle doit également être définie avec un attribut MAXMSGL de 100 Mo.

Remarque : Les programmes d'application écrits par l'utilisateur n'utilisent normalement pas de droits utilisateur de remplacement pour placer des messages dans la file d'attente de messages non livrés. Cela réduit le nombre d'ID utilisateur ayant accès à la file d'attente des messages non livrés.

z/OS *Sécurité de la file d'attente système*

Vous devez configurer l'accès RACF pour autoriser certains ID utilisateur à accéder à des files d'attente système particulières.

La plupart des files d'attente système sont accessibles par les parties auxiliaires de IBM MQ:

- L'utilitaire CSQUTIL
- L'utilitaire de règles de sécurité des messages (CSQ0UTIL)
- Les panneaux d'opérations et de contrôle
- Espace adresse de l'initiateur de canal (y compris le démon de publication / abonnement en file d'attente)
- V 9.1.0 Le serveur mqweb, utilisé par MQ Console et REST API.

Les ID utilisateur sous lesquels ils s'exécutent doivent disposer d'un accès RACF à ces files d'attente, comme indiqué dans la Tableau 35, à la page 210.

Tableau 35. Accès requis aux files d'attente SYSTEM par IBM MQ

File d'attente SYSTEM	CSQUTIL	CSQ0UTIL	Le serveur mqweb	Panneaux d'opérations et de contrôle	Initiateur de canal pour la mise en file d'attente répartie
SYSTEM.ADMIN.CHANNEL.EVENT	-	-	-	-	UPDATE
SYSTEM.ADMIN.COMMAND.QUEUE	-	-	UPDATE	-	-
SYSTEM.BROKER.ADMIN.STREAM	-	-	-	-	ALTER
SYSTEM.BROKER.CONTROL.QUEUE	-	-	-	-	ALTER
SYSTEM.BROKER.DEFAULT.STREAM	-	-	-	-	ALTER
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	-	-	-	-	UPDATE
SYSTEM.CHANNEL.INITQ	-	-	-	-	UPDATE
SYSTEM.CHANNEL.SYNCQ	-	-	-	-	UPDATE
SYSTEM.CLUSTER.COMMAND.QUEUE	-	-	-	-	ALTER
SYSTEM.CLUSTER.REPOSITORY.QUEUE	-	-	-	-	UPDATE
SYSTEM.CLUSTER.TRANSMIT.QUEUE	-	-	-	-	ALTER
SYSTEM.COMMAND.INPUT	UPDATE	-	-	UPDATE	UPDATE
SYSTEM.COMMAND.REPLY.*	-	-	-	-	UPDATE
SYSTEM.COMMAND.REPLY.MODEL	UPDATE	-	-	UPDATE	UPDATE
SYSTEM.CSQOREXX.*	-	-	-	UPDATE	-
SYSTEM.CSQUTIL.*	UPDATE	-	-	-	-
SYSTEM.CSQXCMD.*	-	-	-	-	UPDATE
SYSTEM.HIERARCHY.STATE	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.CONTROL	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.PUBS	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.FANREQ	-	-	-	-	UPDATE
SYSTEM.PROTECTION.ERROR.QUEUE	-	-	-	-	UPDATE

Tableau 35. Accès requis aux files d'attente SYSTEM par IBM MQ (suite)					
File d'attente SYSTEM	**CSQUTIL**	**CSQ0UTIL**	**Le serveur mqweb**	**Panneaux d'opérations et de contrôle**	**Initiateur de canal pour la mise en file d'attente répartie**
SYSTEM.PROTECTION.POLICY.QUEUE	-	Mise à jour«1», à la page 211	-	-	READ
SYSTEM.QSG.CHANNEL.SYNCQ	-	-	-	-	UPDATE
SYSTEM.QSG.TRANSMIT.QUEUE	-	-	-	-	UPDATE
SYSTEM.REST.REPLY.QUEUE	-	-	UPDATE	-	-
SYSTEM.BLUEMIX.REGISTRATION.QUEUE	-	-	-	-	UPDATE

Remarques :

1. L'utilisateur de l'espace adresse Advanced Message Security requiert également un accès en lecture (READ) à cette file d'attente.

z/OS *API-Aide-mémoire sur l'accès à la sécurité des ressources*

Récapitulatif des options **MQOPEN**, **MQPUT1**, **MQSUB**et **MQCLOSE** et de l'accès requis par les différents types de sécurité des ressources.

Tableau 36. Options MQOPEN, MQPUT1, MQSUB et MQCLOSE et autorisation de sécurité requise. Les légendes affichées comme ceci **(1)** font référence aux remarques qui suivent ce tableau.				
	Niveau d'accès RACF minimal requis			
RACF Classe :	**MXTOPIC**	**MQQUEUE ou MXQUEUE (1)**	**MQADMIN ou MXADMIN**	**MQADMIN ou MXADMIN**
RACF profile:	**(15 ou 16)**	**(2)**	**(3)**	**(4)**
Option MQOPEN				
MQOO_INTERROGATION		READ **(5)**	Aucun contrôle	Aucun contrôle
MQOO_BROWSE		READ	Aucun contrôle	Aucun contrôle
MQOO_ENTRÉE_ *		UPDATE	Aucun contrôle	Aucun contrôle
MQOO_SAVE_ALL_CONTEXT **(6)**		UPDATE	Aucun contrôle	Aucun contrôle
MQOO_OUTPUT (UTILISATION = NORMAL) **(7)**		UPDATE	Aucun contrôle	Aucun contrôle
MQOO_PASS_IDENTITY_CONTEXT **(8)**		UPDATE	READ	Aucun contrôle
MQOO_PASS_ALL_CONTEXT **(8) (9)**		UPDATE	READ	Aucun contrôle

Tableau 36. Options MQOPEN, MQPUT1, MQSUB et MQCLOSE et autorisation de sécurité requise. Les légendes affichées comme ceci **(1)** font référence aux remarques qui suivent ce tableau. *(suite)*

	Niveau d'accès RACF minimal requis			
RACF Classe :	**MXTOPIC**	**MQQUEUE ou MXQUEUE (1)**	**MQADMIN ou MXADMIN**	**MQADMIN ou MXADMIN**
RACF profile:	**(15 ou 16)**	**(2)**	**(3)**	**(4)**
MQOO_SET_IDENTITY_CONTEXT **(8) (9)**		UPDATE	UPDATE	Aucun contrôle
MQOO_SET_ALL_CONTEXT **(8) (10)**		UPDATE	CONTROL	Aucun contrôle
MQOO_OUTPUT (SYNTAXE (XMITQ) **(11)**		UPDATE	CONTROL	Aucun contrôle
MQOO_OUTPUT (objet de rubrique)	UPDATE **(16)**			
MQOO_OUTPUT (file d'attente alias vers objet de rubrique)	UPDATE **(16)**	UPDATE		
MQOO_SET		ALTER	Aucun contrôle	Aucun contrôle
MQOO_ALTERNATE_USER_AUTHORITY		**(12)**	**(12)**	UPDATE
Option MQPUT1				
Insertion dans une file d'attente normale **(7)**		UPDATE	Aucun contrôle	Aucun contrôle
MQPMO_PASS_IDENTITY_CONTEXT		UPDATE	READ	Aucun contrôle
MQPMO_PASS_ALL_CONTEXT		UPDATE	READ	Aucun contrôle
MQPMO_SET_IDENTITY_CONTEXT		UPDATE	UPDATE	Aucun contrôle
MQPMO_SET_ALL_CONTEXT		UPDATE	CONTROL	Aucun contrôle
MQOO_SORTIE Insertion dans une file d'attente de transmission **(11)**		UPDATE	CONTROL	Aucun contrôle
MQOO_OUTPUT (objet de rubrique)	UPDATE **(16)**			
MQOO_OUTPUT (file d'attente alias vers objet de rubrique)	UPDATE **(16)**	UPDATE		
MQPMO_ALTERNATE_USER_AUTHORITY		**(13)**	**(13)**	UPDATE
Option MQCLOSE				
MQCO_DELETE **(14)**		ALTER	Aucun contrôle	Aucun contrôle
MQCO_DELETE_PURGE **(14)**		ALTER	Aucun contrôle	Aucun contrôle

Tableau 36. Options MQOPEN, MQPUT1, MQSUB et MQCLOSE et autorisation de sécurité requise. Les légendes affichées comme ceci **(1)** font référence aux remarques qui suivent ce tableau. *(suite)*				
	Niveau d'accès RACF minimal requis			
RACF Classe :	**MXTOPIC**	**MQQUEUE ou MXQUEUE (1)**	**MQADMIN ou MXADMIN**	**MQADMIN ou MXADMIN**
RACF profile:	**(15 ou 16)**	**(2)**	**(3)**	**(4)**
SOUS-TITRE_REMOVE_MQQUE	ALTER **(15)**			
Option MQSUB				
MQSO_CREER	ALTER **(15)**	**(17)**	**(18)**	
MQSO_ALTER	ALTER **(15)**	**(17)**	**(18)**	
MQSO_RESUME	READ **(15)**	**(17)**	Aucun contrôle	
MQSO_ALTERNATE_USER_AUTHORITY				UPDATE
MQSO_SET_IDENTITY_CONTEXT			**(18)**	

Remarque :

1. Cette option n'est pas limitée aux files d'attente. Utilisez la classe MQNLIST ou MXNLIST pour les listes de noms et la classe MQPROC ou MXPROC pour les processus.
2. Utilisez le profil RACF : hlq.resourcename
3. Utilisez le profil RACF : hlq.CONTEXT.queuename
4. Utilisez le profil RACF : hlq.ALTERNATE.USER. `alternateuserid`

   `alternateuserid` est l'identificateur utilisateur spécifié dans la zone *`AlternateUserId`* du descripteur d'objet. Notez que jusqu'à 12 caractères de la zone *`AlternateUserId`* sont utilisés pour cette vérification, contrairement à d'autres vérifications où seuls les 8 premiers caractères d'un identificateur utilisateur sont utilisés.
5. Aucune vérification n'est effectuée lors de l'ouverture du gestionnaire de files d'attente pour les demandes.
6. MQOO_INPUT_ * doit également être spécifié. Valide pour une file d'attente locale, modèle ou alias.
7. Cette vérification est effectuée pour une file d'attente locale ou modèle dont l'attribut de file d'attente **Usage** est MQUS_NORMAL, ainsi que pour un alias ou une file d'attente éloignée (qui est définie pour le gestionnaire de files d'attente connecté). Si la file d'attente est une file d'attente éloignée qui est ouverte en spécifiant explicitement *`ObjectQMgrName`* (et non le nom du gestionnaire de files d'attente connecté), la vérification est effectuée sur la file d'attente portant le même nom que *`ObjectQMgrName`* (qui doit être une file d'attente locale avec l'attribut de file d'attente **Usage** MQUS_TRANSMISSION).
8. MQOO_OUTPUT doit également être spécifié.
9. MQOO_PASS_IDENTITY_CONTEXT est également impliqué par cette option.
10. MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT et MQOO_SET_IDENTITY_CONTEXT sont également impliquées par cette option.
11. Cette vérification est effectuée pour une file d'attente locale ou modèle dont l'attribut de file d'attente **Usage** est MQUS_TRANSMISSION et qui est ouverte directement pour la sortie. Elle ne s'applique pas si une file d'attente éloignée est en cours d'ouverture.
12. Au moins un des paramètres MQOO_INQUIRE, MQOO_BROWSE, MQOO_INPUT_ *, MQOO_OUTPUT ou MQOO_SET doit également être spécifié. Le contrôle effectué est le même que pour les autres options spécifiées.

13. Le contrôle effectué est le même que pour les autres options spécifiées.
14. Cela s'applique uniquement aux files d'attente dynamiques permanentes qui ont été ouvertes directement, c'est-à-dire qui ne sont pas ouvertes via une file d'attente modèle. Aucune sécurité n'est requise pour supprimer une file d'attente dynamique temporaire.
15. Utilisez le profil RACF hlq.SUBSCRIBE.topicname.
16. Utilisez le profil RACF hlq.PUBLISH.topicname.
17. Si, dans la demande MQSUB, vous avez indiqué une file d'attente de destination pour les publications à envoyer, un contrôle de sécurité est effectué sur cette file d'attente pour vous assurer que vous disposez des droits d'insertion sur cette file d'attente.
18. Si dans la demande MQSUB, avec les options MQSO_CREATE ou MQSO_ALTER spécifiées, vous souhaitez définir l'une des zones de contexte d'identité dans la structure MQSD, vous devez également spécifier l'option MQSO_SET_IDENTITY_CONTEXT et vous devez également disposer des droits appropriés sur le profil de contexte de la file d'attente de destination.

z/OS *Profils pour la sécurité des rubriques*

Si la sécurité des rubriques est active, vous devez définir des profils dans les classes appropriées et autoriser les groupes ou les ID utilisateur nécessaires à accéder à ces profils.

Le concept de sécurité des rubriques dans une arborescence de rubriques est décrit dans Sécurité de publication / abonnement.

Si la sécurité des rubriques est active, vous devez effectuer les actions suivantes:

- Définissez des profils dans les classes **MXTOPIC** ou **GMXTOPIC** .
- Autorisez les groupes ou les ID utilisateur nécessaires à accéder à ces profils afin qu'ils puissent émettre des demandes d'API IBM MQ qui utilisent des rubriques.

Les profils de sécurité de rubrique se présentent sous la forme suivante:

```
hlq.SUBSCRIBE.topicname
hlq.PUBLISH.topicname
```

Où

- `hlq` correspond à `qmgr-name` (nom du gestionnaire de files d'attente) ou à `qsg-name` (nom du groupe de partage de files d'attente).
- `topicname` est le nom du noeud d'administration de rubrique dans l'arborescence de rubriques, associé à la rubrique à laquelle il est abonné via un appel MQSUB ou à laquelle il est publié via un appel MQOPEN.

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'accès à une rubrique unique sur ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'accès à une ou plusieurs rubriques avec ce nom de rubrique sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau de gestionnaire de files d'attente pour cette rubrique sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

S'abonner

Pour vous abonner à une rubrique, vous devez accéder à la fois à la rubrique à laquelle vous tentez de vous abonner et à la file d'attente de destination des publications.

Lorsque vous émettez une demande MQSUB, les contrôles de sécurité suivants sont effectués:

- Indique si vous disposez du niveau d'accès approprié pour vous abonner à cette rubrique et si la file d'attente de destination (si spécifiée) est ouverte pour la sortie
- Indique si vous disposez du niveau d'accès approprié à cette file d'attente de destination.

Tableau 37. Niveau d'accès requis pour l'abonnement à la sécurité de rubrique

MQSUB, option	**Accès RACF requis au profil `hlq.SUBSCRIBE.topicname` dans la classe MXTOPIC**
MQSO_CREATE et MQSO_ALTER	ALTER
MQSO_RESUME	READ

Tableau 38. Droits supplémentaires requis pour l'abonnement à l'aide d'une file d'attente de destination non gérée

MQSUB, option	**Accès RACF requis au profil `hlq.CONTEXT.queuename` dans la classe MQADMIN ou MXADMIN**
MQSO_CREATE, MQSO_ALTER et MQSO_RESUME	UPDATE
	AccèsRACF requis au profil `hlq.queuename` dans la classe MQQUEUE ou MXQUEUE
MQSO_CREATE et MQSO_ALTER	UPDATE
	AccèsRACF requis au profil `hlq.ALTERNATE.USER.alternateuserid` dans la classe MQADMIN ou MXADMIN
MQSO_ALTERNATE_USER_AUTHORITY	UPDATE

Remarques sur les files d'attente gérées pour les abonnements

Un contrôle de sécurité est effectué pour voir si vous êtes autorisé à vous abonner à la rubrique. Toutefois, aucun contrôle de sécurité n'est effectué lors de la création de la file d'attente gérée ou pour déterminer si vous avez accès à cette file d'attente de destination pour y placer des messages.

Vous ne pouvez pas fermer la suppression d'une file d'attente gérée.

Les files d'attente modèles utilisées sont: `SYSTEM.DURABLE.MODEL.QUEUE` et `SYSTEM.NDURABLE.MODEL.QUEUE`.

Les files d'attente gérées créées à partir de ces files d'attente modèles sont au format `SYSTEM.MANAGED.DURABLE.A346EF00367849A0` et `SYSTEM.MANAGED.NDURABLE.A346EF0036785EA0` où le dernier qualificateur est imprévisible.

N'accordez aucun accès utilisateur à ces files d'attente. Les files d'attente peuvent être protégées à l'aide de profils génériques au format `SYSTEM.MANAGED.DURABLE.*` et `SYSTEM.MANAGED.NDURABLE.*` sans droits accordés.

Les messages peuvent être extraits de ces files d'attente à l'aide du descripteur renvoyé dans la demande MQSUB.

Si vous émettez explicitement un appel MQCLOSE pour un abonnement avec l'option MQCO_REMOVE_SUB spécifiée et que vous n'avez pas créé l'abonnement que vous fermez sous ce descripteur, un contrôle de sécurité est effectué au moment de la fermeture pour vous assurer que vous disposez des droits appropriés pour effectuer l'opération.

Tableau 39. Niveau d'accès requis aux profils pour la sécurité des rubriques pour la fermeture d'une opération d'abonnement	
MQCLOSE, option	**Accès RACF requis au profil `hlq.SUBSCRIBE.topicname` dans la classe MXTOPIC**
SOUS-TITRE_REMOVE_MQQUE	ALTER

Publication

Pour publier sur une rubrique, vous devez accéder à la rubrique et, si vous utilisez des files d'attente alias, à la file d'attente alias.

Tableau 40. Niveau d'accès requis pour la publication de la sécurité de rubrique	
Option MQOPEN ou MQPUT1	**Accès RACF requis au profil `hlq.PUBLISH.topicname` dans la classe MXTOPIC**
MQOO_OUTPUT ou MQPUT1	UPDATE

Tableau 41. Niveau d'accès requis pour ouvrir une file d'attente alias qui se résout en rubrique	
Option MQOPEN ou MQPUT1	**Accès RACF requis au profil `hlq.queuename` dans la classe MQQUEUE ou MXQUEUE pour la file d'attente alias**
MQOO_OUTPUT ou MQPUT1	UPDATE

Pour plus de détails sur le fonctionnement de la sécurité de rubrique lorsqu'une file d'attente alias qui se résout en un nom de rubrique est ouverte pour publication, voir «Remarques sur les files d'attente alias qui se résolvent en rubriques pour une opération de publication», à la page 216.

Lorsque vous prenez en compte les files d'attente alias utilisées pour les files d'attente de destination pour les restrictions PUT ou GET, voir «Remarques relatives aux files d'attente alias», à la page 205.

Si le niveau d'accès RACF d'une application à un profil de sécurité de rubrique est modifié, les modifications ne prennent effet que pour les nouveaux descripteurs d'objet obtenus (c'est-à-dire, un nouveau MQSUB ou MQOPEN) pour cette rubrique. Ces descripteurs déjà existants au moment de la modification conservent leur accès existant au sujet. En outre, les abonnés existants conservent leur accès à tous les abonnements qu'ils ont déjà souscrits.

Remarques sur les files d'attente alias qui se résolvent en rubriques pour une opération de publication

Lorsque vous émettez un appel MQOPEN ou MQPUT1 pour une file d'attente alias qui se résout en une rubrique, IBM MQ effectue deux vérifications de ressources:

- Premier nom de file d'attente alias indiqué dans le descripteur d'objet (MQOD) dans l'appel MQOPEN ou MQPUT1 .
- Seconde par rapport à la rubrique dans laquelle la file d'attente alias est résolue

Vous devez savoir que ce comportement est différent de celui que vous obtenez lorsque les files d'attente alias sont résolues dans d'autres files d'attente. Vous devez disposer d'un accès correct aux deux profils pour que l'action de publication puisse se poursuivre.

Sécurité des rubriques système

Les rubriques système suivantes sont accessibles par l'espace adresse de l'initiateur de canal.

Les ID utilisateur sous lesquels cette exécution est effectuée doivent disposer d'un accès RACF à ces files d'attente, comme indiqué dans la Tableau 42, à la page 217.

Tableau 42. Accès requis aux rubriques SYSTEM		
Rubrique SYSTEM	**Profil**	**Initiateur de canal pour la mise en file d'attente répartie**
SYSTEM.BROKER.ADMIN.STREAM	hlq.PUBLISH.topicname	UPDATE
SYSTEM.BROKER.ADMIN.STREAM	hlq.SUBSCRIBE.topicname	ALTER

z/OS *Profils pour les processus*

Si la sécurité des processus est active, vous devez définir des profils dans les classes appropriées et autoriser les groupes ou les ID utilisateur nécessaires à accéder à ces profils.

Si la sécurité des processus est active, vous devez:

- Définissez des profils dans les classes **MQPROC** ou **GMQPROC** si vous utilisez des profils en majuscules.
- Définissez des profils dans les classes **MXPROC** ou **GMXPROC** si vous utilisez des profils à casse mixte.
- Autorisez les groupes ou les ID utilisateur nécessaires à accéder à ces profils afin qu'ils puissent émettre des demandes d'API IBM MQ qui utilisent des processus.

Les profils des processus prennent la forme suivante:

```
hlq.processname
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente) et `processname` est le nom du processus en cours d'ouverture.

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'accès à une définition de processus unique sur ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'accès à une ou plusieurs définitions de processus portant ce nom sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau de gestionnaire de files d'attente pour cette définition de processus sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Le tableau suivant présente les droits d'accès requis pour l'ouverture d'un processus.

Tableau 43. Niveaux d'accès pour la sécurité des processus	
MQOPEN, option	**RACF niveau d'accès requis pour hlq.processname**
MQOO_INTERROGATION	READ

Par exemple, sur le gestionnaire de files d'attente MQS9, le groupe RACF INQVPRC doit pouvoir s'interroger (`MQINQ`) sur tous les processus commençant par la lettre V. Les définitions RACF pour cela sont les suivantes:

```
RDEFINE MQPROC MQS9.V* UACC(NONE)
PERMIT MQS9.V* CLASS(MQPROC) ID(INQVPRC) ACCESS(READ)
```

Une autre sécurité utilisateur peut également être active, en fonction des options d'ouverture spécifiées lors de l'ouverture d'un objet de définition de processus.

z/OS *Profils pour les listes de noms*

Si la sécurité de la liste de noms est active, vous définissez des profils dans les classes appropriées et vous accordez aux groupes ou aux ID utilisateur nécessaires l'accès à ces profils.

Si la sécurité de la liste de noms est active, vous devez:

- Définissez des profils dans les classes **MQNLIST** ou **GMQNLIST** si vous utilisez des profils en majuscules.
- Définissez des profils dans les classes **MXNLIST** ou **GMXNLIST** si vous utilisez des profils à casse mixte.
- Autorisez les groupes ou les ID utilisateur nécessaires à accéder à ces profils.

Les profils des listes de noms prennent la forme suivante:

```
hlq.namelistname
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente) et `namelistname` est le nom de la liste de noms ouverte.

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'accès à une liste de noms unique sur ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'accès à une ou plusieurs listes de noms portant ce nom sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau gestionnaire de files d'attente pour cette liste de noms sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Le tableau suivant indique l'accès requis pour l'ouverture d'une liste de noms.

Tableau 44. Niveaux d'accès pour la sécurité de la liste de noms

MQOPEN, option	**RACF niveau d'accès requis pour hlq.namelistname**
MQOO_INTERROGATION	READ

Par exemple, sur le gestionnaire de files d'attente (ou le groupe de partage de files d'attente) PQM3, le groupe RACF DEPT571 doit pouvoir s'interroger (`MQINQ`) sur ces listes de noms:

- Toutes les listes de noms commençant par "DEPT571".
- PRINTER/DESTINATIONS/DEPT571
- AGENCE/DEMANDE/FILES d'attente
- WAREHOUSE.BROADCAST

Les définitions RACF permettant d'effectuer cette opération sont les suivantes:

```
RDEFINE MQNLIST PQM3.DEPT571.** UACC(NONE)
PERMIT PQM3.DEPT571.** CLASS(MQNLIST) ID(DEPT571) ACCESS(READ)

RDEFINE GMQNLIST NLISTS.FOR.DEPT571 UACC(NONE)
        ADDMEM(PQM3.PRINTER/DESTINATIONS/DEPT571,
               PQM3.AGENCY/REQUEST/QUEUES,
               PQM3.WAREHOUSE.BROADCAST)
PERMIT NLISTS.FOR.DEPT571 CLASS(GMQNLIST) ID(DEPT571) ACCESS(READ)
```

Une autre sécurité utilisateur peut être active, selon les options spécifiées lors de l'ouverture d'un objet de liste de noms.

Sécurité de la liste de noms système

La plupart des listes de noms de système sont accessibles par les parties auxiliaires de IBM MQ:

- L'utilitaire CSQUTIL
- Les panneaux d'opérations et de contrôle
- Espace adresse de l'initiateur de canal (y compris le démon de publication / abonnement en file d'attente)

Les ID utilisateur sous lesquels ces exécutions doivent être exécutées doivent disposer d'un accès RACF à ces listes de noms, comme illustré dans la Tableau 45, à la page 219.

Tableau 45. Accès requis aux listes de noms SYSTEM par IBM MQ

Liste de noms SYSTEM	CSQUTIL	Panneaux d'opérations et de contrôle	Initiateur de canal pour la mise en file d'attente répartie
SYSTEM.QPUBSUB.QUEUE.NAMELIST	-	-	READ
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	-	-	READ

z/OS *Profils pour la sécurité des utilisateurs de remplacement*

Si une autre sécurité utilisateur est active, vous devez définir des profils dans les classes appropriées et autoriser les groupes ou les ID utilisateur nécessaires à accéder à ces profils.

Pour plus d'informations sur *AlternateUserId*, voir AlternateUserID (MQCHAR12).

Si la sécurité utilisateur de remplacement est active, vous devez:

- Définissez des profils dans les classes MQADMIN ou GMQADMIN si vous utilisez des profils en majuscules.
- Définissez des profils dans les classes MXADMIN ou GMXADMIN si vous utilisez des profils à casse mixte.

Autorisez les groupes ou les ID utilisateur nécessaires à accéder à ces profils afin qu'ils puissent utiliser les options ALTERNATE_USER_AUTHORITY lorsque l'objet est ouvert.

Les profils de sécurité utilisateur de remplacement peuvent être spécifiés au niveau du sous-système ou du groupe de partage de files d'attente et prendre la forme suivante:

```
hlq.ALTERNATE.USER.alternateuserid
```

Où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente) et `alternateuserid` est la valeur de la zone *AlternateUserId* dans le descripteur d'objet.

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'utilisation d'un autre ID utilisateur sur ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'utilisation d'un autre ID utilisateur sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet ID utilisateur alternatif peut être utilisé sur n'importe quel gestionnaire de files d'attente du groupe de partage de files d'attente par un utilisateur disposant des droits d'accès appropriés. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau gestionnaire de files d'attente pour cet ID utilisateur alternatif sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Le tableau suivant montre l'accès lors de la spécification d'une autre option utilisateur.

Tableau 46. Niveaux d'accès pour la sécurité des utilisateurs de remplacement	
MQOPEN, MQSUB ou MQPUT1	**Niveau d'accès RACF requis**
MQOO_ALTERNATE_USER_AUTHORITY MQSO_ALTERNATE_USER_AUTHORITY MQPMO_ALTERNATE_USER_AUTHORITY	UPDATE

En plus des autres contrôles de sécurité utilisateur, d'autres contrôles de sécurité peuvent également être effectués pour la sécurité de la file d'attente, du processus, de la liste de noms et du contexte. L'ID utilisateur alternatif, s'il est fourni, est utilisé uniquement pour les contrôles de sécurité sur les ressources de file d'attente, de définition de processus ou de liste de noms. Pour les autres vérifications de sécurité d'utilisateur et de contexte, l'ID utilisateur demandant la vérification est utilisé. Pour plus de détails sur la façon dont les ID utilisateur sont gérés, voir «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244. Pour un tableau récapitulatif présentant les options d'ouverture et les contrôles de sécurité requis lorsque la sécurité de la file d'attente, du contexte et de l'utilisateur de remplacement sont tous actifs, voir Tableau 36, à la page 211.

Un autre profil utilisateur permet à l'ID utilisateur demandeur d'accéder aux ressources associées à l'ID utilisateur spécifié dans l'autre ID utilisateur. Par exemple, le serveur de paie s'exécutant sous l'ID utilisateur PAYSERV sur le gestionnaire de files d'attente QMPY traite les demandes des ID utilisateur du personnel, qui commencent tous par PS. Pour que le travail effectué par le serveur de paie soit effectué sous l'ID utilisateur de l'utilisateur demandeur, des droits d'utilisateur alternatifs sont utilisés. Le serveur de paie sait quel ID utilisateur spécifier comme autre ID utilisateur car les programmes demandeurs génèrent des messages à l'aide de l'option de message d'insertion MQPMO_DEFAULT_CONTEXT. Voir «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244 pour plus de détails sur l'emplacement à partir duquel les autres ID utilisateur sont obtenus.

Les exemples de définitions RACF suivants permettent au programme serveur de spécifier d'autres ID utilisateur commençant par les caractères PS:

```
RDEFINE MQADMIN QMPY.ALTERNATE.USER.PS* UACC(NONE)
PERMIT QMPY.ALTERNATE.USER.PS* CLASS(MQADMIN) ID(PAYSERV) ACCESS(UPDATE)
```

Remarque :

1. Les zones *AlternateUserId* du descripteur d'objet et du descripteur d'abonnement ont une longueur de 12 octets. Les 12 octets sont utilisés dans les vérifications de profil, mais seuls les 8 premiers octets sont utilisés comme ID utilisateur par IBM MQ. Si cette troncature d'ID utilisateur n'est pas souhaitable, les programmes d'application qui font la demande doivent convertir tout autre ID utilisateur de plus de 8 octets en quelque chose de plus approprié.
2. Si vous spécifiez MQOO_ALTERNATE_USER_AUTHORITY, MQSO_ALTERNATE_USER_AUTHORITY ou MQPMO_ALTERNATE_USER_AUTHORITY et que vous ne spécifiez pas de zone *AlternateUserId* dans le descripteur d'objet, un ID utilisateur vide est utilisé. Pour les besoins de la vérification de la sécurité de l'utilisateur de remplacement, l'ID utilisateur utilisé pour le qualificateur *AlternateUserId* est -BLANK-. Par exemple, `RDEF MQADMIN hlq.ALTERNATE.USER.-BLANK-`.

   Si l'utilisateur est autorisé à accéder à ce profil, toutes les vérifications supplémentaires sont effectuées avec un ID utilisateur vide. Pour plus de détails sur les ID utilisateur vides, voir «ID utilisateur et niveaux UACC vides», à la page 253.

L'administration des autres ID utilisateur est plus facile si vous disposez d'une convention de dénomination pour les ID utilisateur qui vous permet d'utiliser des profils d'utilisateur alternatifs génériques. Si ce n'est pas le cas, vous pouvez utiliser la fonction RACVARS de RACF . Pour plus d'informations sur l'utilisation de RACVARS, voir le manuel *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

Lorsqu'un message est inséré dans une file d'attente qui a été ouverte avec des droits d'utilisateur de remplacement et que le contexte du message a été généré par le gestionnaire de files d'attente, la zone MQMD_USER_IDENTIFIER est définie sur l'ID utilisateur de remplacement.

z/OS *Profils pour la sécurité de contexte*

IBM MQ utilise des profils pour contrôler l'accès aux informations de contexte spécifiques à un message particulier. Le contexte est contenu dans le descripteur de message (MQMD).

Utilisation de profils pour la sécurité du contexte

Si la sécurité de contexte est active, vous devez:

- Définissez un profil dans la classe **MQADMIN** si vous utilisez des profils en majuscules.
- Définissez le profil dans la classe **MXADMIN** si vous utilisez des profils à casse mixte.

Le profil est appelé `hlq.CONTEXT.queuename` ou `hlq.CONTEXT.topicname`, où:

hlq
Il peut s'agir de `qmgr-name` (nom du gestionnaire de files d'attente) ou de `qsg-name` (nom du groupe de partage de files d'attente).

nom_file_attente
Il peut s'agir du nom complet de la file d'attente pour laquelle vous souhaitez définir le profil de contexte ou d'un profil générique.

topicname
Il peut s'agir du nom complet de la rubrique pour laquelle vous souhaitez définir le profil de contexte ou d'un profil générique.

Un profil préfixé par le nom du gestionnaire de files d'attente et avec ** spécifié comme nom de file d'attente ou de rubrique, permet de contrôler la sécurité contextuelle sur toutes les files d'attente et rubriques appartenant à ce gestionnaire de files d'attente. Il peut être remplacé sur une file d'attente ou une rubrique individuelle en définissant un profil spécifique pour le contexte de cette file d'attente ou rubrique.

Un profil préfixé par le nom du groupe de partage de files d'attente et avec ** spécifié comme nom de file d'attente ou de rubrique, permet de contrôler le contexte sur toutes les files d'attente et les rubriques appartenant aux gestionnaires de files d'attente du groupe de partage de files d'attente. Il peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau gestionnaire de files d'attente pour le contexte sur ce gestionnaire de files d'attente, en spécifiant un profil préfixé par le nom du gestionnaire de files d'attente. Il peut également être remplacé sur une file d'attente ou une rubrique individuelle en spécifiant un profil suffixé avec le nom de la file d'attente ou de la rubrique.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Vous devez accorder aux groupes ou ID utilisateur nécessaires l'accès à ce profil. Le tableau suivant indique le niveau d'accès requis, en fonction de la spécification des options de contexte lors de l'ouverture de la file d'attente.

Tableau 47. Niveaux d'accès pour la sécurité du contexte

Option MQOPEN ou MQPUT1	**Niveau d'accès RACF requis pour hlq.CONTEXT.queuename ou hlq.CONTEXT.topicname**
MQPMO_NO_CONTEXT	Aucun contrôle de sécurité de contexte
MQPMO_CONTEXTE_PAR_DÉFAUT	Aucun contrôle de sécurité de contexte
MQOO_SAVE_ALL_CONTEXT	Aucun contrôle de sécurité de contexte

Tableau 47. Niveaux d'accès pour la sécurité du contexte (suite)	
Option MQOPEN ou MQPUT1	**Niveau d'accès RACF requis pour hlq.CONTEXT.queuename ou hlq.CONTEXT.topicname**
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	READ
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	READ
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	UPDATE
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	CONTROL
MQOO_OUTPUT ou MQPUT1(USAGE (XMITQ))	CONTROL
Option MQSUB	
MQSO_SET_IDENTITY_CONTEXT **(Remarque 2)**	UPDATE

Remarque :

1. Les ID utilisateur utilisés pour la mise en file d'attente répartie requièrent l'accès CONTROL à `hlq.CONTEXT.queuename` pour placer les messages dans la file d'attente de destination. Pour plus d'informations sur les ID utilisateur utilisés, voir «ID utilisateur utilisés par l'initiateur de canal», à la page 248 .
2. Si, dans la demande MQSUB, avec les options MQSO_CREATE ou MQSO_ALTER spécifiées, vous souhaitez définir l'une des zones de contexte d'identité dans la structure MQSD, vous devez spécifier l'option MQSO_SET_IDENTITY_CONTEXT. Vous devez également disposer des droits appropriés sur le profil de contexte de la file d'attente de destination.

Si vous placez des commandes dans la file d'attente d'entrée des commandes système, utilisez l'option de message d'insertion de contexte par défaut pour associer l'ID utilisateur correct à la commande.

Par exemple, le programme utilitaire CSQUTIL fourni par IBM MQpeut être utilisé pour décharger et recharger des messages dans des files d'attente. Lorsque des messages déchargés sont restaurés dans une file d'attente, l'utilitaire CSQUTIL utilise l'option MQOO_SET_ALL_CONTEXT pour rétablir l'état d'origine des messages. En plus de la sécurité de la file d'attente requise par cette option d'ouverture, des droits de contexte sont également requis. Par exemple, si ce droit est requis par le groupe BACKGRP sur le gestionnaire de files d'attente MQS1, il est défini par:

```
RDEFINE MQADMIN MQS1.CONTEXT.** UACC(NONE)
PERMIT MQS1.CONTEXT.** CLASS(MQADMIN) ID(BACKGRP) ACCESS(CONTROL)
```

Selon les options spécifiées et les types de sécurité exécutés, d'autres types de contrôle de sécurité peuvent également se produire lors de l'ouverture de la file d'attente. Il s'agit de la sécurité des files d'attente (voir «Profils pour la sécurité de la file d'attente», à la page 203) et de la sécurité des utilisateurs de remplacement (voir «Profils pour la sécurité des utilisateurs de remplacement», à la page 219). Pour un tableau récapitulatif présentant les options d'ouverture et les contrôles de sécurité requis lorsque la sécurité de la file d'attente, du contexte et de l'utilisateur de remplacement sont tous actifs, voir Tableau 36, à la page 211.

Sécurité du contexte de la file d'attente système

La plupart des files d'attente système sont accessibles par les parties auxiliaires de IBM MQ, par exemple l'espace adresse de l'initiateur de canal V 9.1.0 et le serveur mqweb utilisé par IBM MQ Console et REST API.

Les ID utilisateur sous lesquels ils s'exécutent doivent disposer de l'accès RACF à ces files d'attente, comme indiqué dans Tableau 48, à la page 223.

Tableau 48. Accès requis aux files d'attente SYSTEM pour les opérations de contexte

File d'attente SYSTEM	Initiateur de canal pour la mise en file d'attente répartie	Le serveur mqweb
SYSTEM.ADMIN.COMMAND.QUEUE	-	CONTROL
SYSTEM.BROKER.CONTROL.QUEUE	CONTROL	-
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	CONTROL	-
SYSTEM.CHANNEL.SYNCQ	CONTROL	-
SYSTEM.CLUSTER.COMMAND.QUEUE	CONTROL	-
SYSTEM.CLUSTER.TRANSMIT.QUEUE	CONTROL	-

z/OS *Profils pour la sécurité des commandes*

Pour activer la vérification de la sécurité des commandes, ajoutez des profils à la classe MQCMDS. Les noms de profil sont basés sur les commandes MQSC mais contrôlent à la fois les commandes MQSC et PCF. Les profils peuvent s'appliquer à un gestionnaire de files d'attente ou à un groupe de partage de files d'attente.

Si vous souhaitez effectuer un contrôle de sécurité pour les commandes (de sorte que vous n'ayez pas défini le profil de commutateur de sécurité de commande hlq.NO.CMD.CHECKS) vous devez ajouter des profils à la classe MQCMDS.

Les mêmes profils de sécurité contrôlent les commandes MQSC et PCF. Les noms des profils RACF pour le contrôle de la sécurité des commandes sont basés sur les noms de commande MQSC eux-mêmes. Ces profils se présentent sous la forme suivante:

```
hlq.verb.pkw
```

Où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente), `verb` est la partie instruction du nom de la commande, par exemple ALTER, et `pkw` est le type d'objet, par exemple QLOCAL pour une file d'attente locale.

Par conséquent, le nom de profil de la commande ALTER QLOCAL dans le sous-système CSQ1 est:

```
CSQ1.ALTER.QLOCAL
```

Vous pouvez utiliser des profils génériques pour protéger des ensembles de commandes afin d'avoir moins de profils à gérer et, par conséquent, moins de listes d'accès. Envisagez de créer un profil générique qui s'applique à toutes les commandes non protégées par un profil plus spécifique. Définissez ce profil avec UACC (NONE) et accordez l'accès ALTER uniquement aux groupes RACF contenant des administrateurs. Vous pouvez ensuite créer un profil générique applicable à toutes les commandes DISPLAY et lui accorder un accès étendu. Entre ces extrêmes, vous pouvez identifier des groupes d'utilisateurs ayant besoin d'accéder à certains ensembles de commandes, auquel cas vous pouvez créer des profils pour ces ensembles et accorder l'accès à des groupes RACF représentant ces classes d'utilisateurs. Evitez d'accorder aux utilisateurs l'accès aux commandes dont ils n'ont pas besoin: appliquez le principe du moindre privilège, de sorte que les utilisateurs n'aient accès qu'aux commandes requises pour leurs travaux.

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'utilisation de la commande sur ce gestionnaire de files d'attente. Un profil préfixé par le nom du groupe de partage de files d'attente contrôle l'utilisation de la commande sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en

définissant un profil de niveau de gestionnaire de files d'attente pour cette commande sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

En configurant des profils de commande au niveau du gestionnaire de files d'attente, un utilisateur peut être limité à l'émission de commandes sur un gestionnaire de files d'attente particulier. Vous pouvez également définir un profil pour un groupe de partage de files d'attente pour chaque instruction de commande et tous les contrôles de sécurité sont effectués sur ce profil et non sur des gestionnaires de files d'attente individuels.

Si la sécurité du sous-système et la sécurité du groupe de partage de files d'attente sont toutes deux actives et qu'aucun profil local n'est trouvé, une vérification de la sécurité de la commande est effectuée pour déterminer si l'utilisateur a accès à un profil de groupe de partage de files d'attente.

Si vous utilisez l'attribut CMDSCOPE pour acheminer une commande vers d'autres gestionnaires de files d'attente dans un groupe de partage de files d'attente, la sécurité est vérifiée sur chaque gestionnaire de files d'attente dans lequel la commande est exécutée, mais pas nécessairement sur le gestionnaire de files d'attente dans lequel la commande est entrée.

Le Tableau 49, à la page 224 indique, pour chaque commande IBM MQ MQSC, les profils requis pour la vérification de la sécurité des commandes et le niveau d'accès correspondant pour chaque profil de la classe MQCMDS.

Le Tableau 50, à la page 230 indique, pour chaque commande PCF IBM MQ , les profils requis pour la vérification de la sécurité des commandes et le niveau d'accès correspondant pour chaque profil de la classe MQCMDS.

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés

Commande	**Profil de commande pour MQCMDS**	**Niveau d'accès pour MQCMDS**	**Profil de ressource de commande pour MQADMIN ou MXADMIN**	**Niveau d'accès pour MQADMIN ou MXADMIN**
ALTER AUTHINFO	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
ALTER BUFFPOOL	hlq.ALTER.BUFFPOOL	ALTER	Aucun contrôle	-
ALTER CFSTRUCT	hlq.ALTER.CFSTRUCT	ALTER	Aucun contrôle	-
ALTER CHANNEL	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
ALTER NAMELIST	hlq.ALTER.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
MODIFIER LE PROCESSUS	hlq.ALTER.PROCESS	ALTER	hlq.PROCESS.process	ALTER
ALTER PSID	hlq.ALTER.PSID	ALTER	Aucun contrôle	-
ALTER QALIAS	hlq.ALTER.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
ALTER QLOCAL	hlq.ALTER.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
ALTER QMGR	hlq.ALTER.QMGR	ALTER	Aucun contrôle	-
ALTER QMODEL	hlq.ALTER.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
MODIFIER QREMOTE	hlq.ALTER.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
MODIFIER SECURITE	hlq.ALTER.SECURITY	ALTER	Aucun contrôle	-
ALTER SMDS	hlq.ALTER.SMDS	ALTER	Aucun contrôle	-
ALTER STGCLASS	hlq.ALTER.STGCLASS	ALTER	Aucun contrôle	-
ALTER SUB	hlq.ALTER.SUB	ALTER	Aucun contrôle	-
ALTER TOPIC	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
ALTER TRACE	hlq.ALTER.TRACE	ALTER	Aucun contrôle	-
ARCHIVE LOG	hlq.ARCHIVE.LOG	CONTROL	Aucun contrôle	-
STRUCTURE CFSTRUCT DE SAUVEGARDE	hlq.BACKUP.CFSTRUCT	CONTROL	Aucun contrôle	-
QLOCAL CLEAR	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
CLEAR TOPICSTR «3», à la page 229	hlq.CLEAR.TOPICSTR	ALTER	hlq.TOPIC.topic	ALTER
DEFINE AUTHINFO	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
DEFINIR LE POOL DE MÉMOIRE TAMPON	hlq.DEFINE.BUFFPOOL	ALTER	Aucun contrôle	-
DEFINE CFSTRUCT	hlq.DEFINE.CFSTRUCT	ALTER	Aucun contrôle	-
De la définition d'un canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
DEFINE LOG	hlq.DEFINE.LOG	ALTER	Aucun contrôle	-
DEFINE MAXSMSGS	hlq.DEFINE.MAXSMSGS	ALTER	Aucun contrôle	-
DEFINE NAMELIST	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
DEFINE PROCESSUS	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
DEFINE PSID	hlq.DEFINE.PSID	ALTER	Aucun contrôle	-
DEFINE QALIAS	hlq.DEFINE.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
DEFINE QLOCAL	hlq.DEFINE.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
DEFINIR QMODEL	hlq.DEFINE.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
DEFINIR QREMOTE	hlq.DEFINE.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER
DEFINE STGCLASS	hlq.DEFINE.STGCLASS	ALTER	Aucun contrôle	-
DEFINE SUB	hlq.DEFINE.SUB	ALTER	Aucun contrôle	-
DEFINE TOPIC	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
DELETE AUTHINFO	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
DELETE BUFFPOOL	hlq.DELETE.BUFFPOOL	ALTER	Aucun contrôle	-
DELETE CFSTRUCT	hlq.DELETE.CFSTRUCT	ALTER	Aucun contrôle	-
Supprimer le canal	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Supprimer une liste de noms	hlq.DELETE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Supprimer un processus	hlq.DELETE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
PSID DELETE	hlq.DELETE.PSID	ALTER	Aucun contrôle	-
DELETE QALIAS	hlq.DELETE.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
SUPPRIMER QLOCAL	hlq.DELETE.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
SUPPRIMER QMODEL	hlq.DELETE.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
SUPPRIMER QREMOTE	hlq.DELETE.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER
DELETE STGCLASS	hlq.DELETE.STGCLASS	ALTER	Aucun contrôle	-
SUPPRIMER DES SOUS	hlq.DELETE.SUB	ALTER	Aucun contrôle	-
Supprimer la rubrique	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
AFFICHEZ L'ARCHIVE «1», à la page 229	hlq.DISPLAY.ARCHIVE	READ	Aucun contrôle	-
INFORMATIONS D'AUTHENTIFICATION D'AFFICHAGE	hlq.DISPLAY.AUTHINFO	READ	Aucun contrôle	-
STATUT CF AFFICHAGE	hlq.DISPLAY.CFSTATUS	READ	Aucun contrôle	-
DISPLAY CFSTRUCT	hlq.DISPLAY.CFSTRUCT	READ	Aucun contrôle	-
CANAL D'AFFICHAGE	hlq.DISPLAY.CHANNEL	READ	Aucun contrôle	-
AFFICHAGE DE CHINIT	hlq.DISPLAY.CHINIT	READ	Aucun contrôle	-
AFFICHER CHLAUTH	hlq.DISPLAY.CHLAUTH	READ	Aucun contrôle	-
DISPLAY CHSTATUS	hlq.DISPLAY.CHSTATUS	READ	Aucun contrôle	-
DISPLAY CLUSQMGR	hlq.DISPLAY.CLUSQMGR	READ	Aucun contrôle	-
DISPLAY CMDSERV	hlq.DISPLAY.CMDSERV	READ	Aucun contrôle	-
DISPLAY CONN «1», à la page 229	hlq.DISPLAY.CONN	READ	Aucun contrôle	-
Afficher le groupe	hlq.DISPLAY.GROUP	READ	Aucun contrôle	-

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
JOURNAL D'AFFICHAGE «1», à la page 229	hlq.DISPLAY.LOG	READ	Aucun contrôle	-
DISPLAY MAXSMSGS	hlq.DISPLAY.MAXSMSGS	READ	Aucun contrôle	-
AFFICHAGE DE LA LISTE DE NOMS	hlq.DISPLAY.NAMELIST	READ	Aucun contrôle	-
PROCESSUS D'AFFICHAGE	hlq.DISPLAY.PROCESS	READ	Aucun contrôle	-
AFFICHAGE DE PUBSUB	hlq.DISPLAY.PUBSUB	READ	Aucun contrôle	-
AFFICHER QALIAS	hlq.DISPLAY.QALIAS	READ	Aucun contrôle	-
AFFICHER QCLUSTER	hlq.DISPLAY.QCLUSTER	READ	Aucun contrôle	-
DISPLAY QLOCAL	hlq.DISPLAY.QLOCAL	READ	Aucun contrôle	-
DISPLAY QMGR	hlq.DISPLAY.QMGR	READ	Aucun contrôle	-
AFFICHEZ QMODEL	hlq.DISPLAY.QMODEL	READ	Aucun contrôle	-
AFFICHER QREMOTE	hlq.DISPLAY.QREMOTE	READ	Aucun contrôle	-
STATUT DE LA FILE D'ATTENTE D'AFFICHAGE	hlq.DISPLAY.QSTATUS	READ	Aucun contrôle	-
DISPLAY QUEUE	hlq.DISPLAY.QUEUE	READ	Aucun contrôle	-
STATUT DU JEU DE CARACTÈRES D'AFFICHAGE	hlq.DISPLAY.SBSTATUS	READ	Aucun contrôle	-
Affichage de fichiers de messages partagés (SMDS)	hlq.DISPLAY.SMDS	READ	Aucun contrôle	-
DISPLAY SMDSCONN	hlq.DISPLAY.SMDSCONN	READ	Aucun contrôle	-
DISPLAY SUB	hlq.DISPLAY.SUB	READ	Aucun contrôle	-
SECURITE D'AFFICHAGE	hlq.DISPLAY.SECURITY	READ	Aucun contrôle	-
DISPLAY STGCLASS	hlq.DISPLAY.STGCLASS	READ	Aucun contrôle	-
DISPLAY SYSTEM «1», à la page 229	hlq.DISPLAY.SYSTEM	READ	Aucun contrôle	-
afficher l'unité d'exécution	hlq.DISPLAY.THREAD	READ	Aucun contrôle	-
DISPLAY TPSTATUS	hlq.DISPLAY.TPSTATUS	READ	Aucun contrôle	-

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
DISPLAY TOPIC	hlq.DISPLAY.TOPIC	READ	Aucun contrôle	-
DISPLAY TPSTATUS	hlq.DISPLAY.TPSTATUS	READ	Aucun contrôle	-
AFFICHER TRACE	hlq.DISPLAY.TRACE	READ	Aucun contrôle	-
Affichage d'informations«1», à la page 229	hlq.DISPLAY.USAGE	READ	Aucun contrôle	-
MOVE QLOCAL	hlq.MOVE.QLOCAL	ALTER	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	ALTER
Envoyer une commande PING à un canal	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Récupérer un fichier d'amorce	hlq.RECOVER.BSDS	CONTROL	Aucun contrôle	-
RECUPERER CFSTRUCT	hlq.RECOVER.CFSTRUCT	CONTROL	Aucun contrôle	-
Actualiser le cluster	hlq.REFRESH.CLUSTER	ALTER	Aucun contrôle	-
ACTUALISEZ LE GESTIONNAIRE DE FILES D'ATTENTE	hlq.REFRESH.QMGR	ALTER	Aucun contrôle	-
REFRESH SECURITY	hlq.REFRESH.SECURITY	ALTER	Aucun contrôle	-
REINITIALISER CFSTRUCT	hlq.RESET.CFSTRUCT	CONTROL	Aucun contrôle	-
Réinitialisation du canal	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Réinitialisation d'un cluster	hlq.RESET.CLUSTER	CONTROL	Aucun contrôle	-
RESET QMGR	hlq.RESET.QMGR	CONTROL	Aucun contrôle	-
RESET QSTATS	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
Réinitialiser SMDS	hlq.RESET.SMDS	CONTROL	Aucun contrôle	-
REINITIALISER TPIPE	hlq.RESET.TPIPE	CONTROL	Aucun contrôle	-
Résolution du canal	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Résoudre le statut en attente de validation	hlq.RESOLVE.INDOUBT	CONTROL	Aucun contrôle	-
RESUME QMGR	hlq.RESUME.QMGR	CONTROL	Aucun contrôle	-
SECURITE RVERIFY	hlq.RVERIFY.SECURITY	ALTER	Aucun contrôle	-
Définir une archive	hlq.SET.ARCHIVE	CONTROL	Aucun contrôle	-

Tableau 49. Commandes MQSC, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
SET CHLAUTH	hlq.SET.CHLAUTH	CONTROL	Aucun contrôle	-
Définir un journal	hlq.SET.LOG	CONTROL	Aucun contrôle	-
Définir un système	hlq.SET.SYSTEM	CONTROL	Aucun contrôle	-
Démarrer un canal	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
START CHINIT «4», à la page 230	hlq.START.CHINIT	CONTROL	Aucun contrôle	-
DEMARRAGE de CMDSERV	hlq.START.CMDSERV	CONTROL	Aucun contrôle	-
Démarrer le programme d'écoute	hlq.START.LISTENER	CONTROL	Aucun contrôle	-
START QMGR	AUCUN«2», à la page 229	-	-	-
DEMARRAGE de SMDSCONN	hlq.START.SMDSCONN	CONTROL	Aucun contrôle	-
Démarrer une trace	hlq.START.TRACE	CONTROL	Aucun contrôle	-
Arrêter le canal	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
STOP CHINIT	hlq.STOP.CHINIT	CONTROL	Aucun contrôle	-
ARRETER CMDSERV	hlq.STOP.CMDSERV	CONTROL	Aucun contrôle	-
Arrêter le programme d'écoute	hlq.STOP.LISTENER	CONTROL	Aucun contrôle	-
STOP QMGR	hlq.STOP.QMGR	CONTROL	Aucun contrôle	-
ARRETER SMDSCONN	hlq.STOP.SMDSCONN	CONTROL	Aucun contrôle	-
Arrêter le traçage	hlq.STOP.TRACE	CONTROL	Aucun contrôle	-
SUSPEND QMGR	hlq.SUSPEND.QMGR	CONTROL	Aucun contrôle	-

Remarques :

1. Ces commandes peuvent être émises en interne par le gestionnaire de files d'attente ; aucun droit n'est vérifié dans ces cas.
2. IBM MQ ne vérifie pas les droits de l'utilisateur qui exécute la commande START QMGR. Toutefois, vous pouvez utiliser RACFou vos autres fonctions de sécurité pour contrôler l'accès à la commande START xxxxMSTR émise à la suite de la commande START QMGR. Pour ce faire, vous pouvez contrôler l'accès au profil MVS.START.STC.xxxxMSTR dans la classe des commandes de l'opérateur RACF (OPERCMDS). Pour plus d'informations sur cette procédure, voir le manuel *z/OS SecureWay Security Server RACF Security Administrator's Guide*. Si vous utilisez cette technique et qu'un utilisateur non autorisé tente de démarrer le gestionnaire de files d'attente, il s'arrête avec le code anomalie 00F30216.
3. La ressource **hlq.TOPIC.topic** fait référence à l'objet Topic dérivé de TOPICSTR. Pour plus de détails, voir «Sécurité de publication / abonnement», à la page 480

4. Dans les versions antérieures à IBM MQ for z/OS V6, le contrôle de sécurité portait sur MVS.START.STC.CSQ1CHIN. A partir de la version IBM MQ for z/OS V6 , le nom de la ressource est associé à un qualificateur JOBNAME supplémentaire. Cela peut entraîner des problèmes lors du démarrage de l'initiateur de canal.

   Pour résoudre le problème, remplacez MVS.START.STC. *ssid* CHIN avec un profil pour une ressource nommée MVS.START.STC. *ssid* CHIN **.*** ou MVS.START.STC. *ssid* CHIN. *ssid* CHIN où *ssid* est l'ID de sous-système du gestionnaire de files d'attente. Pour cela, vous devez disposer des droits d'accès RACF UPDATE. Pour plus de détails, voir Documentation du produit z/OS for *Operation planning, MVS Commands, RACF Access Authorities, and Resource Names*.

   Le paramètre START for *ssid* MSTR n'inclut pas le paramètre JOBNAME=. Par souci de cohérence, vous pouvez mettre à jour le profil de MVS.START.STC.ssidMSTR en MVS.START.STC.ssidMSTR. *.

Tableau 50. Commandes PCF, profils et niveaux d'accès associés

Commande	**Profil de commande pour MQCMDS**	**Niveau d'accès pour MQCMDS**	**Profil de ressource de commande pour MQADMIN ou MXADMIN**	**Niveau d'accès pour MQADMIN ou MXADMIN**
Sauvegarder une structure d'unité de couplage	hlq.BACKUP.CFSTRUCT	CONTROL	Aucun contrôle	-
Modifier l'objet d'informations d'authentification	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Modifier une structure d'unité de couplage	hlq.ALTER.CFSTRUCT	ALTER	Aucun contrôle	-
Modifier un canal	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Modifier une liste de noms	hlq.ALTER.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Changement de processus	hlq.ALTER.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Modifier une file d'attente	hlq.ALTER.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Modifier un gestionnaire de files d'attente	hlq.ALTER.QMGR	ALTER	Aucun contrôle	-
Modifier la sécurité	hlq.ALTER.SECURITY	ALTER	Aucun contrôle	-
Modifier SMDS	hlq.ALTER.SMDS	ALTER	Aucun contrôle	-
Modifier une classe d'archivage	hlq.ALTER.STGCLASS	ALTER	Aucun contrôle	-
Modifier un abonnement	hlq.ALTER.SUB	ALTER	Aucun contrôle	-
Modifier une rubrique	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Mettre à blanc une file d'attente	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
Effacer la chaîne d'une rubrique«1», à la page 235	hlq.CLEAR.TOPICSTR	ALTER	hlq.TOPIC.topic	ALTER
Copier l'objet d'informations d'authentification	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER

Tableau 50. Commandes PCF, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
Copier une structure d'unité de couplage	hlq.DEFINE.CFSTRUCT	ALTER	Aucun contrôle	-
Copier un canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Copier une liste de noms	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Copier un processus	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Copier une file d'attente	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Copier un abonnement	hlq.DEFINE.SUB	ALTER	Aucun contrôle	-
Copier une classe d'archivage	hlq.DEFINE.STGCLASS	ALTER	Aucun contrôle	-
Copier une rubrique	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Créer un objet d'informations d'authentification	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Créer une structure d'unité de couplage	hlq.DEFINE.CFSTRUCT	ALTER	Aucun contrôle	-
Créer un canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Créer une liste de noms	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Créer un processus	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Créer une file d'attente	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Créer une création de classe d'archivage	hlq.DEFINE.STGCLASS	ALTER	Aucun contrôle	-
Créer un abonnement	hlq.DEFINE.SUB	ALTER	Aucun contrôle	-
Créer une rubrique	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Supprimer l'objet d'informations d'authentification	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Supprimer une structure d'unité de couplage	hlq.DELETE.CFSTRUCT	ALTER	Aucun contrôle	-
Supprimer un canal	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Supprimer une liste de noms	hlq.DELETE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Supprimer un processus	hlq.DELETE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Supprimer une file d'attente	hlq.DELETE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Supprimer une classe d'archivage	hlq.DELETE.STGCLASS	ALTER	Aucun contrôle	-
Supprimer l'abonnement	hlq.DELETE.SUB	ALTER	Aucun contrôle	-
Supprimer une rubrique	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Consulter une archive	hlq.DISPLAY.ARCHIVE	READ	Aucun contrôle	-

Tableau 50. Commandes PCF, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
Consulter l'objet d'informations d'authentification	hlq.DISPLAY.AUTHINFO	READ	Aucun contrôle	-
Consulter les noms d'objet d'informations d'authentification	hlq.DISPLAY.AUTHINFO	READ	Aucun contrôle	-
Consulter une structure d'unité de couplage	hlq.DISPLAY.CFSTRUCT	READ	Aucun contrôle	-
Consulter des noms de structure d'unité de couplage	hlq.DISPLAY.CFSTRUCT	READ	Aucun contrôle	-
Consulter le statut d'une structure d'unité de couplage	hlq.DISPLAY.CFSTATUS	READ	Aucun contrôle	-
Consulter un canal	hlq.DISPLAY.CHANNEL	READ	Aucun contrôle	-
Consulter les enregistrements d'authentification de canal	hlq.DISPLAY.CHLAUTH	READ	Aucun contrôle	-
Consulter l'initiateur de canal	hlq.DISPLAY.CHINIT	READ	Aucun contrôle	-
Consulter les noms de canal	hlq.DISPLAY.CHANNEL	READ	Aucun contrôle	-
Consulter le statut d'un canal	hlq.DISPLAY.CHSTATUS	READ	Aucun contrôle	-
Consulter un gestionnaire de files d'attente de cluster	hlq.DISPLAY.CLUSQMGR	READ	Aucun contrôle	-
Consulter une connexion	hlq.DISPLAY.CONNPCF	READ	Aucun contrôle	-
Interroger le groupe	hlq.DISPLAY.GROUP	READ	Aucun contrôle	-
Consulter un journal	hlq.DISPLAY.LOG	READ	Aucun contrôle	-
Consulter une liste de noms	hlq.DISPLAY.NAMELIST	READ	Aucun contrôle	-
Consulter les noms de liste de noms	hlq.DISPLAY.NAMELIST	READ	Aucun contrôle	-
Consulter un processus	hlq.DISPLAY.PROCESS	READ	Aucun contrôle	-
Consulter les noms de processus	hlq.DISPLAY.PROCESS	READ	Aucun contrôle	-
Consulter le statut de publication / d'abonnement	hlq.DISPLAY.PUBSUB	READ	Aucun contrôle	-
Consulter la file d'attente	hlq.DISPLAY.QUEUE	READ	Aucun contrôle	-

Tableau 50. Commandes PCF, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
Consulter les gestionnaires de files d'attente	hlq.DISPLAY.QMGR	READ	Aucun contrôle	-
Consulter les noms de file d'attente	hlq.DISPLAY.QUEUE	READ	Aucun contrôle	-
Consulter le statut d'une file d'attente	hlq.DISPLAY.QSTATUS	READ	Aucun contrôle	-
Consulter la sécurité	hlq.DISPLAY.SECURITY	READ	Aucun contrôle	-
Consulter SMDS	hlq.DISPLAY.SMDS	READ	Aucun contrôle	-
Consulter SMDSCONN	hlq.DISPLAY.SMDSCONN	READ	Aucun contrôle	-
Consulter une classe d'archivage	hlq.DISPLAY.STGCLASS	READ	Aucun contrôle	-
Consulter des noms de classe d'archivage	hlq.DISPLAY.STGCLASS	READ	Aucun contrôle	-
Consulter un abonnement	hlq.INQUIRE.SUB	READ	Aucun contrôle	-
Consulter le statut de l'abonnement	hlq.INQUIRE.SBSTATUS	READ	Aucun contrôle	-
Consulter un système	hlq.DISPLAY.SYSTEM	READ	Aucun contrôle	-
Consulter une rubrique	hlq.DISPLAY.TOPIC	READ	Aucun contrôle	-
Consulter les noms de rubrique	hlq.DISPLAY.TOPIC	READ	Aucun contrôle	-
Consulter le statut d'une rubrique	hlq.DISPLAY.TPSTATUS	READ	Aucun contrôle	-
Consulter une utilisation	hlq.DISPLAY.USAGE	READ	Aucun contrôle	-
Déplacer une file d'attente	hlq.MOVE.QLOCAL	ALTER	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	ALTER
Envoyer une commande PING à un canal	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Récupérer une structure d'unité de couplage	hlq.RECOVER.CFSTRUCT	CONTROL	Aucun contrôle	-
Régénérer un cluster	hlq.REFRESH.CLUSTER	ALTER	Aucun contrôle	-
Régénérer un gestionnaire de files d'attente	hlq.REFRESH.QMGR	ALTER	Aucun contrôle	-
Régénérer la sécurité	hlq.REFRESH.SECURITY	ALTER	Aucun contrôle	-
Réinitialiser la structure CF	hlq.RESET.CFSTRUCT	CONTROL	Aucun contrôle	-
Réinitialisation du canal	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Réinitialisation d'un cluster	hlq.RESET.CLUSTER	CONTROL	Aucun contrôle	-

Tableau 50. Commandes PCF, profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
Réinitialiser un gestionnaire de files d'attente	hlq.RESET.QMGR	CONTROL	Aucun contrôle	-
Réinitialiser les statistiques de file d'attente	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
Réinitialiser SMDS	hlq.RESET.SMDS	CONTROL	Aucun contrôle	-
Résolution du canal	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Reprendre un gestionnaire de files d'attente	hlq.RESUME.QMGR	CONTROL	Aucun contrôle	-
Reprendre un cluster de gestionnaire de files d'attente	hlq.RESUME.QMGR	CONTROL	Aucun contrôle	-
Revérifier la sécurité	hlq.RVERIFY.SECURITY	ALTER	Aucun contrôle	-
Définir une archive	hlq.SET.ARCHIVE	CONTROL	Aucun contrôle	-
Définir l'enregistrement d'authentification de canal	hlq.SET.CHLAUTH	CONTROL	Aucun contrôle	-
Définir un journal	hlq.SET.LOG	CONTROL	Aucun contrôle	-
Définir un système	hlq.SET.SYSTEM	CONTROL	Aucun contrôle	-
Démarrer un canal	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Démarrer un initialisateur de canal	hlq.START.CHINIT	CONTROL	Aucun contrôle	-
Démarrer un programme d'écoute de canaux	hlq.START.LISTENER	CONTROL	Aucun contrôle	-
Démarrer la connexion SMDS	hlq.START.SMDSCONN	CONTROL	Aucun contrôle	-
Arrêter le canal	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Arrêter l'initialisateur de canal	hlq.STOP.CHINIT	CONTROL	Aucun contrôle	-
Arrêter un programme d'écoute de canaux	hlq.STOP.LISTENER	CONTROL	Aucun contrôle	-
Arrêter la connexion SMDS	hlq.STOP.SMDSCONN	CONTROL	Aucun contrôle	-
Interrompre un gestionnaire de files d'attente	hlq.SUSPEND.QMGR	CONTROL	Aucun contrôle	-
Interrompre un cluster de gestionnaire de files d'attente	hlq.SUSPEND.QMGR	CONTROL	Aucun contrôle	-

Remarques :

1. La ressource **hlq.TOPIC.topic** fait référence à l'objet Topic dérivé de TOPICSTR. Pour plus de détails, voir «Sécurité de publication / abonnement», à la page 480

V 9.1.0 Pour plus de détails sur les profils PCF IBM MQ requis lors de l'utilisation du IBM MQ Console, voir «IBM MQ Console -profils de sécurité de commande requis», à la page 235 .

z/OS V 9.1.0 *IBM MQ Console -profils de sécurité de commande requis*

Les opérations effectuées dans le IBM MQ Console par un utilisateur du rôle `MQWebAdmin`ou `MQWebAdminRO`sont effectuées dans le contexte de sécurité de l'ID utilisateur de la tâche démarrée du serveur mqweb. Si vous souhaitez utiliser IBM MQ Console, l'ID utilisateur de la tâche démarrée du serveur mqweb doit être autorisé à émettre certaines commandes PCF.

Tableau 51, à la page 235 affiche, pour chaque commande PCF IBM MQ , les profils de sécurité de commande requis et le niveau d'accès correspondant pour chaque profil de la classe MQCMDS requis par IBM MQ Console.

Tableau 51. Commandes PCF IBM MQ Console , profils et niveaux d'accès associés

Commande	**Profil de commande pour MQCMDS**	**Niveau d'accès pour MQCMDS**	**Profil de ressource de commande pour MQADMIN ou MXADMIN**	**Niveau d'accès pour MQADMIN ou MXADMIN**
Modifier l'objet d'informations d'authentification	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Modifier un canal	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Modifier une file d'attente	hlq.ALTER.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Modifier un gestionnaire de files d'attente	hlq.ALTER.QMGR	ALTER	Aucun contrôle	-
Modifier une rubrique	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Mettre à blanc une file d'attente	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
Créer un objet d'informations d'authentification	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Créer un canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Créer une file d'attente	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Créer un abonnement	hlq.DEFINE.SUB	ALTER	Aucun contrôle	-
Créer une rubrique	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Supprimer l'objet d'informations d'authentification	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Supprimer un canal	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Supprimer une file d'attente	hlq.DELETE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Supprimer l'abonnement	hlq.DELETE.SUB	ALTER	Aucun contrôle	-
Supprimer une rubrique	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Consulter l'objet d'informations d'authentification	hlq.DISPLAY.AUTHINFO	READ	Aucun contrôle	-

Tableau 51. Commandes PCF IBM MQ Console , profils et niveaux d'accès associés (suite)

Commande	Profil de commande pour MQCMDS	Niveau d'accès pour MQCMDS	Profil de ressource de commande pour MQADMIN ou MXADMIN	Niveau d'accès pour MQADMIN ou MXADMIN
Consulter les noms d'objet d'informations d'authentification	hlq.DISPLAY.AUTHINFO	READ	Aucun contrôle	-
Consulter un canal	hlq.DISPLAY.CHANNEL	READ	Aucun contrôle	-
Consulter les enregistrements d'authentification de canal	hlq.DISPLAY.CHLAUTH	READ	Aucun contrôle	-
Consulter l'initiateur de canal	hlq.DISPLAY.CHINIT	READ	Aucun contrôle	-
Consulter les noms de canal	hlq.DISPLAY.CHANNEL	READ	Aucun contrôle	-
Consulter le statut d'un canal	hlq.DISPLAY.CHSTATUS	READ	Aucun contrôle	-
Consulter la file d'attente	hlq.DISPLAY.QUEUE	READ	Aucun contrôle	-
Consulter les gestionnaires de files d'attente	hlq.DISPLAY.QMGR	READ	Aucun contrôle	-
Consulter les noms de file d'attente	hlq.DISPLAY.QUEUE	READ	Aucun contrôle	-
Consulter le statut d'une file d'attente	hlq.DISPLAY.QSTATUS	READ	Aucun contrôle	-
Consulter un abonnement	hlq.INQUIRE.SUB	READ	Aucun contrôle	-
Consulter le statut de l'abonnement	hlq.INQUIRE.SBSTATUS	READ	Aucun contrôle	-
Consulter une rubrique	hlq.DISPLAY.TOPIC	READ	Aucun contrôle	-
Consulter les noms de rubrique	hlq.DISPLAY.TOPIC	READ	Aucun contrôle	-
Consulter le statut d'une rubrique	hlq.DISPLAY.TPSTATUS	READ	Aucun contrôle	-
Envoyer une commande PING à un canal	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Régénérer un cluster	hlq.REFRESH.CLUSTER	ALTER	Aucun contrôle	-
Régénérer la sécurité	hlq.REFRESH.SECURITY	ALTER	Aucun contrôle	-
Réinitialisation du canal	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Résolution du canal	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Définir l'enregistrement d'authentification de canal	hlq.SET.CHLAUTH	CONTROL	Aucun contrôle	-
Démarrer un canal	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Arrêter le canal	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL

z/OS *Profils pour la sécurité des ressources de commande*

Si vous n'avez pas défini le profil de commutateur de sécurité de ressource de commande, vous devez ajouter des profils de ressource pour chaque ressource à la classe appropriée, car vous souhaitez vérifier la sécurité des ressources associées aux commandes. Les mêmes profils de sécurité contrôlent les commandes MQSC et PCF.

Si vous n'avez pas défini le profil de commutateur de sécurité de ressource de commande, `hlq.NO.CMD.RESC.CHECKS`, car vous souhaitez vérifier la sécurité des ressources associées aux commandes, vous devez:

- Ajoutez un profil de ressource dans la classe **MQADMIN** , si vous utilisez des profils en majuscules, pour chaque ressource.
- Ajoutez un profil de ressource dans la classe **MXADMIN** , si vous utilisez des profils à casse mixte, pour chaque ressource.

Les mêmes profils de sécurité contrôlent les commandes MQSC et PCF.

Les profils pour la vérification de la sécurité des ressources de commande se présentent sous la forme suivante:

```
hlq.type.resourcename
```

où `hlq` peut être `qmgr-name` (nom du gestionnaire de files d'attente) ou `qsg-name` (nom du groupe de partage de files d'attente).

Un profil préfixé par le nom du gestionnaire de files d'attente contrôle l'accès aux ressources associées aux commandes de ce gestionnaire de files d'attente. Un profil préfixé par le nom de groupe de partage de files d'attente contrôle l'accès aux ressources associées aux commandes sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cet accès peut être remplacé sur un gestionnaire de files d'attente individuel en définissant un profil de niveau gestionnaire de files d'attente pour cette ressource de commande sur ce gestionnaire de files d'attente.

Si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous utilisez la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ recherche d'abord un profil préfixé par le nom du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil préfixé par le nom du groupe de partage de files d'attente.

Par exemple, le nom de profil RACF pour la vérification de la sécurité de la ressource de commande par rapport à la file d'attente modèle CREDIT.WORTHY dans le sous-système CSQ1 est:

```
CSQ1.QUEUE.CREDIT.WORTHY
```

Etant donné que les profils de tous les types de ressource de commande sont conservés dans la classe MQADMIN, la partie "type" du nom de profil est nécessaire dans le profil pour faire la distinction entre les ressources de types différents ayant le même nom. La partie "type" du nom de profil peut être CHANNEL, QUEUE, TOPIC, PROCESS ou NAMELIST. Par exemple, un utilisateur peut être autorisé à définir hlq.QUEUE.PAYROLL.ONE, mais non autorisé à définir hlq.PROCESS.PAYROLL.ONE

Si le type de ressource est une file d'attente et que le profil est un profil de niveau groupe de partage de files d'attente, il contrôle l'accès à une ou plusieurs files d'attente locales du groupe de partage de files d'attente ou l'accès à une file d'attente partagée unique à partir de n'importe quel gestionnaire de files d'attente du groupe de partage de files d'attente.

z/OS Les commandes MQSC, les profils et leurs niveaux d'accès affichent, pour chaque commande IBM MQ MQSC, les profils requis pour la vérification de la sécurité des commandes et le niveau d'accès correspondant pour chaque profil de la classe MQCMDS.

z/OS Les commandes PCF, les profils et leurs niveaux d'accès affichent, pour chaque commande PCF IBM MQ , les profils requis pour la vérification de la sécurité des commandes à exécuter, ainsi que le niveau d'accès correspondant pour chaque profil de la classe MQCMDS.

z/OS *Contrôle de la sécurité des ressources de commande pour les files d'attente alias et les files d'attente distantes*

Les files d'attente alias et les files d'attente éloignées fournissent toutes deux une indirection vers une autre file d'attente. Des points supplémentaires s'appliquent lorsque vous envisagez de vérifier la sécurité de ces files d'attente.

Files d'attente alias

Lorsque vous définissez une file d'attente alias, les contrôles de sécurité des ressources de commande sont effectués uniquement sur le nom de la file d'attente alias et non sur le nom de la file d'attente cible dans laquelle l'alias est résolu.

Les files d'attente alias peuvent être résolues en files d'attente locales et éloignées. Si vous ne souhaitez pas autoriser les utilisateurs à accéder à certaines files d'attente locales ou distantes, vous devez effectuer les deux opérations suivantes:

1. N'autorisez pas les utilisateurs à accéder à ces files d'attente locales et éloignées.
2. Empêche les utilisateurs de pouvoir définir des alias pour ces files d'attente. C'est-à-dire qu'ils ne peuvent pas émettre les commandes DEFINE QALIAS et ALTER QALIAS.

Files d'attente éloignées

Lorsque vous définissez une file d'attente éloignée, les contrôles de sécurité des ressources de commande ne sont effectués que sur le nom de la file d'attente éloignée. Aucune vérification n'est effectuée sur les noms des files d'attente spécifiées dans les attributs RNAME ou XMITQ de la définition d'objet de file d'attente éloignée.

z/OS Profil de sécurité RESLEVEL

Vous pouvez définir un profil spécial dans la classe MQADMIN ou MXADMIN pour contrôler le nombre d'ID utilisateur vérifiés pour la sécurité des ressources d'API. Ce profil est appelé profil RESLEVEL. La manière dont ce profil affecte la sécurité des ressources d'API dépend de la manière dont vous accédez à IBM MQ.

Lorsqu'une application tente de se connecter à IBM MQ, IBM MQ vérifie l'accès que l'ID utilisateur associé à la connexion a à un profil dans la classe MQADMIN ou MXADMIN appelé:

```
hlq.RESLEVEL
```

Où `hlq` peut être `ssid` (ID de sous-système) ou `qsg` (ID de groupe de partage de files d'attente).

Les ID utilisateur associés à chaque type de connexion sont les suivants:

- ID utilisateur de la tâche de connexion pour les connexions par lots
- ID utilisateur de l'espace adresse CICS pour les connexions CICS
- ID utilisateur de l'espace adresse de la région IMS pour les connexions IMS
- ID utilisateur de l'espace adresse de l'initiateur de canal pour les connexions de l'initiateur de canal

Avertissement : RESLEVEL est une option très puissante ; elle peut entraîner le contournement de toutes les vérifications de la sécurité des ressources pour une connexion particulière.

Si aucun profil RESLEVEL n'est défini, vous devez veiller à ce qu'aucun autre profil de la classe MQADMIN ne corresponde à hlq.RESLEVEL. Par exemple, si vous avez un profil dans MQADMIN appelé hlq. * * et pas de profil hlq.RESLEVEL , méfiez-vous des conséquences de hlq. * * car il est utilisé pour la vérification RESLEVEL.

Définissez un profil hlq.RESLEVEL et définissez UACC sur NONE, au lieu de n'avoir aucun profil RESLEVEL. Avoir le moins d'utilisateurs ou de groupes possible dans la liste d'accès. Pour plus de détails sur l'audit de l'accès RESLEVEL, voir «Remarques sur l'audit sous z/OS», à la page 265.

Si vous utilisez uniquement la sécurité au niveau du gestionnaire de files d'attente, IBM MQ effectue des vérifications RESLEVEL sur le profil `qmgr-name.RESLEVEL` . Si vous utilisez uniquement la sécurité au niveau du groupe de partage de files d'attente, IBM MQ effectue des vérifications RESLEVEL sur le profil `qsg-name.RESLEVEL` . Si vous utilisez une combinaison de la sécurité au niveau du gestionnaire de files d'attente et du groupe de partage de files d'attente, IBM MQ vérifie d'abord l'existence d'un profil RESLEVEL au niveau du gestionnaire de files d'attente. S'il n'en trouve pas, il recherche un profil RESLEVEL au niveau du groupe de partage de files d'attente.

S'il ne trouve pas de profil RESLEVEL, IBM MQ active la vérification de l'ID du travail et de la tâche (ou de l'utilisateur de remplacement) pour une connexion CICS ou IMS . Pour une connexion par lots, IBM MQ active la vérification de l'ID utilisateur du travail (ou d'un autre ID utilisateur). Pour l'initiateur de canal, IBM MQ active la vérification de l'ID utilisateur du canal et de l'ID utilisateur MCA (ou autre).

S'il existe un profil RESLEVEL, le niveau de vérification dépend de l'environnement et du niveau d'accès du profil.

N'oubliez pas que si votre gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente et que vous ne définissez pas ce profil au niveau du gestionnaire de files d'attente, il se peut qu'un profil défini au niveau du groupe de partage de files d'attente affecte le niveau de checking.To activer la vérification de deux ID utilisateur, vous devez définir un profil RESLEVEL (précédé du nom du gestionnaire de files d'attente du nom du groupe de partage de files d'attente) avec une valeur UACC (NONE) et vous assurer que les utilisateurs concernés ne disposent pas des droits d'accès à ce profil.

Lorsque vous prenez en compte l'accès de l'ID utilisateur de l'initiateur de canal à RESLEVEL, n'oubliez pas que la connexion établie par l'initiateur de canal est également la connexion utilisée par les canaux. Un paramètre qui provoque le contournement de tous les contrôles de sécurité de ressource pour l'ID utilisateur de l'initiateur de canal ignore les contrôles de sécurité pour tous les canaux. Si l'accès de l'ID utilisateur de l'initiateur de canal à RESLEVEL est différent de NONE, un seul ID utilisateur (pour un niveau d'accès READ ou UPDATE) ou aucun ID utilisateur (pour un niveau d'accès CONTROL ou ALTER) est vérifié pour l'accès. Si vous accordez à l'ID utilisateur de l'initiateur de canal un niveau d'accès autre que NONE à RESLEVEL, assurez-vous que vous comprenez l'effet de ce paramètre sur les contrôles de sécurité effectués pour les canaux.

L'utilisation du profil RESLEVEL signifie que les enregistrements d'audit de sécurité normaux ne sont pas utilisés. Par exemple, si vous placez UAUDIT sur un utilisateur, l'accès au profil hlq.RESLEVEL dans MQADMIN n'est pas audité.

Si vous utilisez l'option RACF WARNING sur le profil hlq.RESLEVEL , aucun message d'avertissement RACF n'est généré pour les profils de la classe RESLEVEL.

La vérification de la sécurité pour les messages de rapport tels que les COD est contrôlée par le profil RESLEVEL associé à l'application d'origine. Par exemple, si l'ID utilisateur d'un travail par lots dispose des droits CONTROL ou ALTER sur un profil RESLEVEL, toutes les vérifications de ressources effectuées par le travail par lots sont ignorées, y compris la vérification de la sécurité des messages de rapport.

Si vous modifiez le profil RESLEVEL, les utilisateurs doivent se déconnecter et se reconnecter avant que la modification n'ait lieu. (Cela inclut l'arrêt et le redémarrage de l'initiateur de canal si l'accès de l'ID utilisateur de l'espace adresse de mise en file d'attente répartie au profil RESLEVEL est modifié.)

Pour désactiver l'audit RESLEVEL, utilisez le paramètre système RESAUDIT.

z/OS *RESLEVEL et connexions par lots*

Par défaut, lorsqu'une ressource IBM MQ est accessible via des connexions par lots et de type lot, l'utilisateur doit être autorisé à accéder à cette ressource pour l'opération particulière. Vous pouvez ignorer le contrôle de sécurité en configurant une définition RESLEVEL appropriée.

La vérification de l'utilisateur dépend de l'ID utilisateur utilisé lors de la connexion, c'est-à-dire de l'ID utilisateur utilisé pour la vérification de la connexion.

Par exemple, vous pouvez configurer RESLEVEL de sorte que lorsqu'un utilisateur auquel vous faites confiance accède à certaines ressources via une connexion par lots, aucun contrôle de sécurité de ressource d'API n'est effectué ; mais lorsqu'un utilisateur auquel vous ne faites pas confiance tente d'accéder aux mêmes ressources, les contrôles de sécurité sont effectués normalement. Vous devez

configurer la vérification RESLEVEL pour ignorer les vérifications de la sécurité des ressources d'API uniquement lorsque vous faites suffisamment confiance à l'utilisateur et aux programmes exécutés par cet utilisateur.

Le tableau suivant présente les vérifications effectuées pour les connexions par lots.

Tableau 52. Vérifications effectuées à différents niveaux d'accès RACF pour les connexions par lots

RACF Niveau d'accès	**Niveau de contrôle**
AUCUN	Vérifications de ressources effectuées
READ	Vérifications de ressources effectuées
UPDATE	Vérifications de ressources effectuées
CONTROL	Aucune vérification.
ALTER	Aucune vérification.

z/OS *RESLEVEL et les fonctions système*

Application de RESLEVEL aux panneaux d'exploitation et de contrôle et à CSQUTIL.

Les panneaux d'opération et de contrôle et l'utilitaire CSQUTIL sont des applications de type lot qui envoient des demandes au serveur de commandes du gestionnaire de files d'attente. Ils sont donc soumis aux remarques décrites dans «RESLEVEL et connexions par lots», à la page 239. Vous pouvez utiliser RESLEVEL pour ignorer la vérification de la sécurité pour SYSTEM.COMMAND.INPUT et SYSTEM.COMMAND.REPLY.MODEL qu'ils utilisent, mais pas pour les files d'attente dynamiques SYSTEM.CSQXCMD. *, SYSTEM.CSQOREXX.*, et SYSTEM.CSQUTIL. *.

Le serveur de commandes fait partie intégrante du gestionnaire de files d'attente et n'est donc pas associé à une connexion ou à une vérification RESLEVEL. Par conséquent, pour maintenir la sécurité, le serveur de commandes doit confirmer que l'ID utilisateur de l'application à l'origine de la demande est autorisé à ouvrir la file d'attente utilisée pour les réponses. Pour les opérations et les panneaux de commande, il s'agit de SYSTEM.CSQOREXX. *. Pour CSQUTIL, il s'agit de SYSTEM.CSQUTIL. *. Les utilisateurs doivent être autorisés à utiliser ces files d'attente, comme décrit dans «Sécurité de la file d'attente système», à la page 209, en plus de toute autorisation RESLEVEL qui leur est accordée.

Pour les autres applications utilisant le serveur de commandes, il s'agit de la file d'attente de réponse qu'elles nomment. Ces autres applications peuvent tromper le serveur de commandes en plaçant des messages dans des files d'attente non autorisées en transmettant (dans le contexte du message) un ID utilisateur plus digne de confiance que le sien au serveur de commandes. Pour éviter cela, utilisez un profil CONTEXT pour protéger le contexte d'identité des messages placés dans SYSTEM.COMMAND.INPUT.

z/OS *Connexions RESLEVEL et CICS*

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué sur une connexion CICS , deux ID utilisateur sont vérifiés. Vous pouvez modifier les ID utilisateur à vérifier en configurant un profil RESLEVEL.

Le premier ID utilisateur vérifié est celui de l'espace adresse CICS . Il s'agit de l'ID utilisateur sur la carte de travail du travail CICS ou de l'ID utilisateur affecté à la tâche démarrée CICS par la classe z/OS STARTED ou la table des procédures démarrées. (Il ne s'agit pas de CICS DFLTUSER.)

Le deuxième ID utilisateur vérifié est l'ID utilisateur associé à la transaction CICS .

Si l'un de ces ID utilisateur n'a pas accès à la ressource, la demande échoue avec le code achèvement MQRC_NOT_AUTHORIZED. L'ID utilisateur de l'espace adresse CICS et l'ID utilisateur de la personne exécutant la transaction CICS doivent avoir accès à la ressource au niveau approprié.

Comment RESLEVEL peut affecter les vérifications effectuées

Selon la manière dont vous configurez votre profil RESLEVEL, vous pouvez modifier les ID utilisateur qui sont vérifiés lorsque l'accès à une ressource est demandé. Pour plus d'informations, voir Tableau 53, à la page 241.

Les ID utilisateur vérifiés dépendent de l'ID utilisateur utilisé lors de la connexion, c'est-à-dire l'ID utilisateur de l'espace adresse CICS . Ce contrôle permet de contourner la vérification de la sécurité des ressources d'API pour les demandes IBM MQ provenant d'un système (par exemple, un système de test, TESTCICS), mais de les implémenter pour un autre (par exemple, un système de production, PRODCICS).

Remarque : Si vous configurez votre ID utilisateur d'espace adresse CICS avec l'attribut "trusted" dans la classe STARTED ou la RACF table des procédures démarrées ICHRIN03, cela remplace toute vérification d'ID utilisateur pour l'espace adresse CICS établie par le profil RESLEVEL pour votre gestionnaire de files d'attente (c'est-à-dire que le gestionnaire de files d'attente n'effectue pas les vérifications de sécurité pour l'espace adresse CICS). Pour en savoir davantage, reportez-vous au manuel *CICS Transaction Server for z/OS V3.2 RACF -Guide de sécurité*.

Le tableau suivant présente les vérifications effectuées pour les connexions CICS .

Tableau 53. Vérifications effectuées à différents niveaux d'accès RACF pour les connexions CICS

RACF Niveau d'accès	**Niveau de contrôle**
AUCUN	IBM MQ vérifie l'ID utilisateur de l'espace adresse CICS et l'ID utilisateur de la transaction.
READ	IBM MQ vérifie uniquement l'ID utilisateur de l'espace adresse CICS .
UPDATE	Si la transaction est définie sur CICS avec RESSEC (YES), IBM MQ vérifie l'ID utilisateur de l'espace adresse CICS et l'ID utilisateur de la transaction.
UPDATE	Si la transaction est définie sur CICS avec RESSEC (NO), IBM MQ vérifie uniquement l'ID utilisateur de l'espace adresse CICS .
CONTROL ou ALTER	IBM MQ ne vérifie aucun ID utilisateur.

z/OS *Connexions RESLEVEL et IMS*

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué pour une connexion IMS , deux ID utilisateur sont vérifiés. Vous pouvez modifier les ID utilisateur à vérifier en configurant un profil RESLEVEL.

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué pour une connexion IMS , deux ID utilisateur sont vérifiés pour déterminer si l'accès à la ressource est autorisé.

Le premier ID utilisateur vérifié est celui de l'espace adresse de la région IMS . Cette valeur provient de la zone USER de la carte de travail ou de l'ID utilisateur affecté à la région à partir de la classe z/OS STARTED ou de la table des procédures démarrées (SPT).

Le deuxième ID utilisateur vérifié est associé au travail effectué dans la région dépendante. Il est déterminé en fonction du type de la région dépendante, comme indiqué dans Comment le deuxième ID utilisateur est déterminé pour la connexion IMS(tm).

Si le premier ou le deuxième ID utilisateur IMS n'a pas accès à la ressource, la demande échoue avec le code achèvement MQRC_NOT_AUTHORIZED.

Le paramètre des profils IBM MQ RESLEVEL ne peut pas modifier l'ID utilisateur sous lequel les transactions IMS sont planifiées à partir du programme de moniteur de déclenchement CSQQTRMN MQ-IMS fourni par IBM. Cet ID utilisateur est le PSBNAME de ce moniteur de déclenchement, qui est par défaut CSQQTRMN.

Comment RESLEVEL peut affecter les vérifications effectuées

Selon la manière dont vous configurez votre profil RESLEVEL, vous pouvez modifier les ID utilisateur qui sont vérifiés lorsque l'accès à une ressource est demandé. Les vérifications possibles sont les suivantes:

- Vérifiez l'ID utilisateur de l'espace adresse de la région IMS et le deuxième ID utilisateur ou l'ID utilisateur de remplacement.
- Vérifiez uniquement l'ID utilisateur de l'espace adresse de la région IMS .
- Ne vérifiez aucun ID utilisateur.

Le tableau suivant présente les vérifications effectuées pour les connexions IMS .

Tableau 54. Vérifications effectuées à différents niveaux d'accès RACF pour les connexions IMS

RACF Niveau d'accès	Niveau de contrôle
AUCUN	Vérifiez l'ID utilisateur de l'espace adresse IMS et le deuxième ID utilisateur ou l'ID utilisateur de remplacement IMS .
READ	Vérifiez l'ID utilisateur de l'espace adresse IMS .
UPDATE	Vérifiez l'ID utilisateur de l'espace adresse IMS .
CONTROL	Aucune vérification.
ALTER	Aucune vérification.

z/OS *RESLEVEL et la connexion de l'initiateur de canal*

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué par l'initiateur de canal, deux ID utilisateur sont vérifiés. Vous pouvez modifier les ID utilisateur à vérifier en configurant un profil RESLEVEL.

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué par l'initiateur de canal, deux ID utilisateur sont vérifiés pour déterminer si l'accès à la ressource est autorisé.

Les ID utilisateur vérifiés peuvent être ceux spécifiés par l'attribut de canal MCAUSER, ceux reçus du réseau, ceux de l'espace adresse de l'initiateur de canal ou ceux de l'autre ID utilisateur pour le descripteur de message. Les ID utilisateur vérifiés dépendent du protocole de communication que vous utilisez et de la définition de l'attribut de canal PUTAUT. Pour plus d'informations, voir «ID utilisateur utilisés par l'initiateur de canal», à la page 248.

Si l'un de ces ID utilisateur n'a pas accès à la ressource, la demande échoue avec le code achèvement MQRC_NOT_AUTHORIZED.

Comment RESLEVEL peut affecter les vérifications effectuées

En fonction de la manière dont vous avez configuré votre profil RESLEVEL, vous pouvez modifier les ID utilisateur qui sont vérifiés lorsque l'accès à une ressource est demandé et le nombre de ceux qui sont vérifiés.

Le tableau suivant montre les vérifications effectuées pour la connexion de l'initiateur de canal et pour tous les canaux car ils utilisent cette connexion.

Tableau 55. Vérifications effectuées à différents niveaux d'accès RACF pour les connexions d'initiateur de canal

RACF Niveau d'accès	Niveau de contrôle
AUCUN	Vérifiez deux ID utilisateur.
READ	Vérifiez un ID utilisateur.
UPDATE	Vérifiez un ID utilisateur.

Tableau 55. Vérifications effectuées à différents niveaux d'accès RACF pour les connexions d'initiateur de canal (suite)	
RACF Niveau d'accès	**Niveau de contrôle**
CONTROL	Aucune vérification.
ALTER	Aucune vérification.
Remarque : Voir «ID utilisateur utilisés par l'initiateur de canal», à la page 248 pour une définition des ID utilisateur vérifiés	

z/OS *RESLEVEL et la mise en file d'attente intra-groupe*

Par défaut, lorsqu'un contrôle de sécurité de ressource d'API est effectué par l'agent de mise en file d'attente intra-groupe, deux ID utilisateur sont vérifiés pour déterminer si l'accès à la ressource est autorisé. Vous pouvez modifier les ID utilisateur à vérifier en configurant un profil RESLEVEL.

Les ID utilisateur vérifiés peuvent être l'ID utilisateur déterminé par l'attribut IGQUSER du gestionnaire de files d'attente de réception, l'ID utilisateur du gestionnaire de files d'attente au sein du groupe de partage de files d'attente qui a inséré le message dans SYSTEM.QSG.TRANSMIT.QUEUE, ou l'autre ID utilisateur spécifié dans la zone *UserIdentifier* du descripteur de message du message. Pour plus d'informations, voir «ID utilisateur utilisés par l'agent de mise en file d'attente intra-groupe», à la page 252.

L'agent de mise en file d'attente intra-groupe étant une tâche de gestionnaire de files d'attente interne, il n'émet pas de demande de connexion explicite et s'exécute sous l'ID utilisateur du gestionnaire de files d'attente. L'agent de mise en file d'attente intra-groupe démarre lors de l'initialisation du gestionnaire de files d'attente. Lors de l'initialisation de l'agent de mise en file d'attente intra-groupe, IBM MQ vérifie l'accès de l'ID utilisateur associé au gestionnaire de files d'attente à un profil de la classe MQADMIN appelé:

```
hlq.RESLEVEL
```

Cette vérification est toujours effectuée sauf si le commutateur hlq.NO.SUBSYS.SECURITY a été défini.

S'il n'existe pas de profil RESLEVEL, IBM MQ permet de vérifier deux ID utilisateur. S'il existe un profil RESLEVEL, le niveau de vérification dépend du niveau d'accès accordé à l'ID utilisateur du gestionnaire de files d'attente pour le profil. Les vérifications effectuées à différents niveaux d'accès RACF(r) pour l'agent de mise en file d'attente intra-groupe montrent les vérifications effectuées pour l'agent de mise en file d'attente intra-groupe.

Tableau 56. Vérifications effectuées à différents niveaux d'accès RACF pour l'agent de mise en file d'attente intragroupe	
RACF Niveau d'accès	**Niveau de contrôle**
AUCUN	Vérifiez deux ID utilisateur.
READ	Vérifiez un ID utilisateur.
UPDATE	Vérifiez un ID utilisateur.
CONTROL	Aucune vérification.
ALTER	Aucune vérification.
Remarque : Voir «ID utilisateur utilisés par l'agent de mise en file d'attente intra-groupe», à la page 252 pour une définition des ID utilisateur vérifiés	

Si les droits accordés au profil RESLEVEL pour l'ID utilisateur du gestionnaire de files d'attente sont modifiés, l'agent de mise en file d'attente intra-groupe doit être arrêté et redémarré pour récupérer les

nouveaux droits. Comme il n'est pas possible d'arrêter et de redémarrer indépendamment l'agent de mise en file d'attente intra-groupe, le gestionnaire de files d'attente doit être arrêté et redémarré pour ce faire.

z/OS *RESLEVEL et les ID utilisateur vérifiés*

Exemple de définition d'un profil RESLEVEL et d'octroi d'un accès à ce profil.

Vérification de l'ID utilisateur par rapport au nom de profil pour les connexions par lots via Vérification des ID utilisateur par rapport au nom de profil pour la LU 6.2 et les canaux de connexion au serveur TCP/IP montre comment RESLEVEL affecte les ID utilisateur vérifiés pour les différentes demandes MQI.

Par exemple, vous disposez d'un gestionnaire de files d'attente appelé QM66 avec les exigences suivantes:

- L'utilisateur WS21B doit être exempté de la sécurité des ressources.
- La tâche démarrée CICS WXNCICS s'exécutant sous l'ID utilisateur d'espace adresse CICSWXN doit effectuer une vérification complète des ressources uniquement pour les transactions définies avec RESSEC (YES).

Pour définir le profil RESLEVEL approprié, exécutez la commande RACF suivante:

```
RDEFINE MQADMIN QM66.RESLEVEL UACC(NONE)
```

Accordez ensuite aux utilisateurs l'accès à ce profil à l'aide des commandes suivantes:

```
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(WS21B) ACCESS(CONTROL)
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(CICSWXN) ACCESS(UPDATE)
```

Si vous effectuez ces modifications alors que les ID utilisateur sont connectés au gestionnaire de files d'attente QM66, les utilisateurs doivent se déconnecter et se reconnecter avant que la modification ne soit effectuée.

Si la sécurité du sous-système n'est pas active lorsqu'un utilisateur se connecte, mais que cet utilisateur est toujours connecté, la sécurité du sous-système devient active, la vérification de la sécurité complète des ressources est appliquée à l'utilisateur. L'utilisateur doit se reconnecter pour obtenir le traitement RESLEVEL correct.

z/OS ID utilisateur pour le contrôle de sécurité sous z/OS

IBM MQ lance des contrôles de sécurité basés sur les ID utilisateur associés à des utilisateurs, des terminaux, des applications et d'autres ressources. Cette collection de rubriques répertorie les ID utilisateur utilisés pour chaque type de contrôle de sécurité.

z/OS *ID utilisateur pour la sécurité de la connexion*

L'ID utilisateur utilisé pour la sécurité de connexion dépend du type de connexion.

Type de connexion	Contenu de l'ID utilisateur
Connexion par lots	ID utilisateur de la tâche de connexion. Exemple : • ID utilisateur TSO • ID utilisateur affecté à un travail par lots par le paramètre USER JCL • ID utilisateur affecté à une tâche démarrée par la classe STARTED ou la table des procédures démarrées
Connexion CICS	ID utilisateur de l'espace adresse CICS .
Connexion IMS	ID utilisateur de l'espace adresse de la région IMS .

Type de connexion	Contenu de l'ID utilisateur
Connexion de l'initiateur de canal	ID utilisateur de l'espace adresse de l'initiateur de canal.

z/OS *ID utilisateur pour la sécurité des commandes et des ressources de commandes*

L'ID utilisateur utilisé pour la sécurité de la commande ou de la ressource de la commande dépend de l'emplacement à partir duquel la commande est émise.

Délivré à partir de ...	Contenu de l'ID utilisateur
CSQINP1, CSQINP2ou CSQINPT	Aucune vérification n'est effectuée.
File d'attente d'entrée des commandes système	ID utilisateur trouvé dans le *UserIdentifier* du descripteur de message du message qui contient la commande. Si le message ne contient pas de *UserIdentifier*, un ID utilisateur vide est transmis au gestionnaire de sécurité.
Console	ID utilisateur connecté à la console. Si la console n'est pas connectée, l'ID utilisateur par défaut défini par le paramètre système CMDUSER dans CSQ6SYSP. Pour émettre des commandes à partir d'une console, la console doit disposer de l'attribut z/OS SYS AUTHORITY.
Console SDSF/TSO	TSO ou ID utilisateur du travail.
Panneaux d'opérations et de contrôle	ID utilisateur TSO. Si vous prévoyez d'utiliser les panneaux d'opérations et de contrôle, vous devez disposer des droits appropriés pour exécuter les commandes correspondant aux actions que vous choisissez. En outre, vous devez disposer d'un accès en lecture à tous les hlq.DISPLAY. Les profils *objet* dans la classe MQCMDS car les panneaux utilisent les différentes commandes DISPLAY pour collecter les informations qu'ils présentent.
MGCRE	Si MGCRE est utilisé avec UTOKEN, ID utilisateur dans UTOKEN. Si MGCRE est émis sans UTOKEN, l'ID utilisateur TSO ou de travail est utilisé.
CSQ0UTIL	ID utilisateur du travail.
CSQUTIL	ID utilisateur du travail.
CSQINPX	ID utilisateur de l'espace adresse de l'initiateur de canal.

z/OS *ID utilisateur pour la sécurité des ressources (MQOPEN, MQSUB et MQPUT1)*

Ces informations affichent le contenu des ID utilisateur pour les ID utilisateur normaux et de remplacement pour chaque type de connexion. Le nombre de vérifications est défini par le profil RESLEVEL. L'ID utilisateur vérifié est celui utilisé pour les appels **MQOPEN**, **MQSUB**ou **MQPUT1** .

Remarque : Toutes les zones d'ID utilisateur sont vérifiées exactement comme elles sont reçues. Aucune conversion n'a lieu et, par exemple, trois zones d'ID utilisateur contenant "Bob", "BOB"et "bob" ne sont pas équivalentes.

z/OS *ID utilisateur vérifiés pour les connexions par lots*

L'ID utilisateur vérifié pour une connexion par lots dépend de la façon dont la tâche est exécutée et de la spécification d'un autre ID utilisateur.

Tableau 57. Vérification de l'ID utilisateur par rapport au nom de profil pour les connexions par lots			
ID utilisateur de remplacement spécifié à l'ouverture?	**hlq.ALTERNATE.USER.userid**	**Profil hlq.CONTEXT.queuename**	**Profil hlq.resourcename**
Non	-	JOB	JOB
Yes	JOB	JOB	ALT

Clé :

ALT
Autre ID utilisateur.

JOB

- ID utilisateur d'une connexion TSO ou USS.
- ID utilisateur affecté à un travail par lots.
- ID utilisateur affecté à une tâche démarrée par la classe STARTED ou la table des procédures démarrées.
- ID utilisateur associé à la procédure mémorisée Db2 en cours d'exécution

Un travail par lots exécute un MQPUT1 dans une file d'attente appelée Q1 avec RESLEVEL défini sur READ et la vérification de l'ID utilisateur de remplacement désactivée.

Les vérifications effectuées à différents niveaux d'accès RACF(r) pour les connexions par lots et la vérification de l'ID utilisateur par rapport au nom de profil pour les connexions par lots indiquent que l'ID utilisateur du travail est vérifié par rapport au profil hlq.Q1.

z/OS *ID utilisateur vérifiés pour les connexions CICS*
Les ID utilisateur vérifiés pour les connexions CICS dépendent de l'exécution d'une ou de deux vérifications et de la spécification d'un autre ID utilisateur.

Tableau 58. Vérification de l'ID utilisateur par rapport au nom de profil pour les ID utilisateur CICS-type			
ID utilisateur de remplacement spécifié à l'ouverture?	**hlq.ALTERNATE.USER.userid**	**Profil hlq.CONTEXT.queuename**	**Profil hlq.resourcename**
Non, 1 vérification	-	ADS	ADS
Non, 2 vérifications	-	ADS + TXN	ADS + TXN
Oui, 1 vérification	ADS	ADS	ADS
Oui, 2 vérifications	ADS + TXN	ADS + TXN	ADS + ALT

Clé :

ALT
ID utilisateur de remplacement

ADS
ID utilisateur associé au travail par lots CICS ou, si CICS s'exécute en tant que tâche démarrée, via la classe STARTED ou la table des procédures démarrées.

TXN
ID utilisateur associé à la transaction CICS . Il s'agit normalement de l'ID utilisateur de l'utilisateur de terminal qui a démarré la transaction. Il peut s'agir de l'utilisateur CICS DFLTUSER, d'un terminal de sécurité PRESET ou d'un utilisateur connecté manuellement.

Déterminez les ID utilisateur vérifiés pour les conditions suivantes:

- Le niveau d'accès RACF au profil RESLEVEL, pour un ID utilisateur d'espace adresse CICS , est défini sur NONE.
- Un appel MQOPEN est effectué sur une file d'attente avec MQOO_OUTPUT et MQOO_PASS_IDENTITY_CONTEXT.

Commencez par voir combien d'ID utilisateur CICS sont vérifiés en fonction de l'accès de l'ID utilisateur de l'espace adresse CICS au profil RESLEVEL. Depuis Tableau 53, à la page 241 dans la rubrique «Connexions RESLEVEL et CICS», à la page 240, deux ID utilisateur sont vérifiés si le profil RESLEVEL est défini sur NONE. Ensuite, à partir de Tableau 58, à la page 246 , ces vérifications sont effectuées:

- hlq.ALTERNATE.USER.userid n'est pas vérifié.
- Le profil hlq.CONTEXT.queuename est vérifié avec l'ID utilisateur de l'espace adresse CICS et l'ID utilisateur de la transaction CICS .
- Le profil hlq.resourcename est vérifié avec l'ID utilisateur de l'espace adresse CICS et l'ID utilisateur de la transaction CICS .

Cela signifie que quatre contrôles de sécurité sont effectués pour cet appel MQOPEN .

z/OS *ID utilisateur vérifiés pour les connexions IMS*

Les ID utilisateur vérifiés pour les connexions IMS dépendent de l'exécution d'une ou de deux vérifications et de la spécification d'un autre ID utilisateur. Si un deuxième ID utilisateur est vérifié, il dépend du type de région dépendante et des ID utilisateur disponibles.

Tableau 59. Vérification de l'ID utilisateur par rapport au nom de profil pour les ID utilisateur IMS-type

ID utilisateur de remplacement spécifié à l'ouverture?	hlq.ALTERNATE.USER.userid	Profil hlq.CONTEXT.queuename	Profil hlq.resourcename
Non, 1 vérification	-	REG	REG
Non, 2 vérifications	-	REG + SEC	REG + SEC
Oui, 1 vérification	REG	REG	REG
Oui, 2 vérifications	REG + SEC	REG + SEC	REG + ALT

Clé :

ALT
Autre ID utilisateur.

REG
L'ID utilisateur est normalement défini via la classe STARTED ou la table des procédures démarrées ou, si IMS est en cours d'exécution, à partir d'un travail soumis, par le paramètre USER JCL.

SEC
Le deuxième ID utilisateur est associé au travail effectué dans une région dépendante. Il est déterminé en fonction de Tableau 60, à la page 247.

Tableau 60. Comment le deuxième ID utilisateur est déterminé pour la connexion IMS

Types de région dépendante	Hiérarchie de détermination du deuxième ID utilisateur
• Message BMP géré et GET UNIQUE réussi émis. • IFP et GET UNIQUE émises. • MPP.	ID utilisateur associé à la transaction IMS si l'utilisateur est connecté. Nom LTERM, s'il est disponible. PSBNAME.

Tableau 60. Comment le deuxième ID utilisateur est déterminé pour la connexion IMS (suite)	
Types de région dépendante	**Hiérarchie de détermination du deuxième ID utilisateur**
• Message BMP géré et GET UNIQUE réussie non émis. • BMP n'est pas géré par message. • IFP et GET UNIQUE non émis.	ID utilisateur associé à l'espace adresse de la région dépendante IMS s'il ne s'agit pas de blancs ou de zéros. PSBNAME.

z/OS *ID utilisateur utilisés par l'initiateur de canal*

Cette collection de rubriques décrit les ID utilisateur utilisés et vérifiés pour la réception des canaux et pour les demandes MQI client émises via les canaux de connexion serveur. Des informations sont fournies pour TCP/IP et pour LU6.2

Vous pouvez utiliser le paramètre PUTAUT de la définition de canal récepteur pour déterminer le type de contrôle de sécurité utilisé. Pour obtenir un contrôle de sécurité cohérent dans votre réseau IBM MQ , vous pouvez utiliser les options ONLYMCA et ALTMCA.

Vous pouvez utiliser la commande DISPLAY CHSTATUS pour déterminer l'ID utilisateur utilisé par l'agent MCA.

z/OS *Réception de canaux à l'aide de TCP/IP*

Les ID utilisateur vérifiés dépendent de l'option PUTAUT du canal et de l'exécution d'une ou de deux vérifications.

Tableau 61. ID utilisateur vérifiés par rapport au nom de profil pour les canaux TCP/IP			
Option PUTAUT indiquée sur le canal récepteur ou demandeur	**hlq.ALTERNATE.USER.userid**	**Profil hlq.CONTEXT.queuename**	**Profil hlq.resourcename**
DEF, 1 vérification	-	CHL	CHL
DEF, 2 vérifications	-	CHL + MCA	CHL + MCA
CTX, 1 vérification	CHL	CHL	CHL
CTX, 2 vérifications	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 vérification	-	MCA	MCA
ONLYMCA, 2 vérifications	-	MCA	MCA
ALTMCA, 1 vérification	MCA	MCA	MCA
ALTMCA, 2 vérifications	MCA	MCA	MCA + ALT

Clé :

MCA (ID utilisateur MCA)
ID utilisateur indiqué pour l'attribut de canal MCAUSER au niveau du récepteur ; s'il est vide, l'ID utilisateur de l'espace adresse de l'initiateur de canal du côté du récepteur ou du demandeur est utilisé.

CHL (ID utilisateur de canal)
Sur TCP/IP, la sécurité n'est pas prise en charge par le système de communication du canal. Si Transport Layer Security (TLS) est utilisé et qu'un certificat numérique a été transmis par le partenaire, l'ID utilisateur associé à ce certificat (s'il est installé) ou l'ID utilisateur associé à un filtre correspondant trouvé à l'aide de RACF Certificate Name Filtering (CNF) est utilisé. Si aucun ID utilisateur associé n'est trouvé ou si TLS n'est pas utilisé, l'ID utilisateur de l'espace adresse de l'initiateur de canal de l'extrémité du récepteur ou du demandeur est utilisé comme ID utilisateur de canal sur les canaux définis avec le paramètre PUTAUT défini sur DEF ou CTX.

Remarque : L'utilisation de RACF Certificate Name Filtering (CNF) vous permet d'affecter le même ID utilisateur RACF à plusieurs utilisateurs distants, par exemple tous les utilisateurs de la même unité organisationnelle, qui possèdent naturellement tous les mêmes droits de sécurité. Cela signifie que le serveur n'a pas besoin de disposer d'une copie du certificat de chaque utilisateur distant possible à travers le monde, et simplifie considérablement la gestion et la distribution des certificats.

Si le paramètre PUTAUT est défini sur ONLYMCA ou ALTMCA pour le canal, l'ID utilisateur du canal est ignoré et l'ID utilisateur MCA du récepteur ou du demandeur est utilisé. Cela s'applique également aux canaux TCP/IP utilisant TLS.

ALT (ID utilisateur de remplacement)
ID utilisateur provenant des informations de contexte (c'est-à-dire la zone *UserIdentifier*) dans le descripteur de message du message. Cet ID utilisateur est déplacé dans la zone *AlternateUserID* du descripteur d'objet avant qu'un appel **MQOPEN** ou **MQPUT1** ne soit émis pour la file d'attente de destination cible.

z/OS *Réception de canaux à l'aide de LU 6.2*
Les ID utilisateur vérifiés dépendent de l'option PUTAUT du canal et de l'exécution d'une ou de deux vérifications.

Tableau 62. ID utilisateur vérifiés par rapport au nom de profil pour les canaux LU 6.2

Option PUTAUT indiquée sur le canal récepteur ou demandeur	hlq.ALTERNATE.USER.userid	Profil hlq.CONTEXT.queuename	Profil hlq.resourcename
DEF, 1 vérification	-	CHL	CHL
DEF, 2 vérifications	-	CHL + MCA	CHL + MCA
CTX, 1 vérification	CHL	CHL	CHL
CTX, 2 vérifications	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 vérification	-	MCA	MCA
ONLYMCA, 2 vérifications	-	MCA	MCA
ALTMCA, 1 vérification	MCA	MCA	MCA
ALTMCA, 2 vérifications	MCA	MCA	MCA + ALT

Clé :

MCA (ID utilisateur MCA)
ID utilisateur indiqué pour l'attribut de canal MCAUSER au niveau du récepteur ; s'il est vide, l'ID utilisateur de l'espace adresse de l'initiateur de canal du côté du récepteur ou du demandeur est utilisé.

CHL (ID utilisateur de canal)

Canaux demandeur-serveur

Si le canal est démarré à partir du demandeur, il n'est pas possible de recevoir un ID utilisateur réseau (ID utilisateur du canal).

Si le paramètre PUTAUT est défini sur DEF ou CTX sur le canal demandeur, l'ID utilisateur du canal est celui de l'espace adresse de l'initiateur de canal du demandeur car aucun ID utilisateur n'est reçu du réseau.

Si le paramètre PUTAUT est défini sur ONLYMCA ou ALTMCA, l'ID utilisateur du canal est ignoré et l'ID utilisateur MCA du demandeur est utilisé.

Autres types de canaux

Si le paramètre PUTAUT est défini sur DEF ou CTX sur le canal récepteur ou demandeur, l'ID utilisateur du canal correspond à l'ID utilisateur reçu du système de communication lors du lancement du canal.

- Si le canal émetteur est sous z/OS, l'ID utilisateur du canal reçu est l'ID utilisateur de l'espace adresse de l'initiateur de canal de l'émetteur.
- Si le canal émetteur se trouve sur une autre plateforme (par exemple, AIX), l'ID utilisateur de canal reçu est généralement fourni par le paramètre USERID de la définition de canal.

Si l'ID utilisateur reçu est vide ou qu'aucun ID utilisateur n'est reçu, un ID utilisateur de canal vide est utilisé.

ALT (ID utilisateur de remplacement)

ID utilisateur provenant des informations de contexte (c'est-à-dire la zone *UserIdentifier*) dans le descripteur de message du message. Cet ID utilisateur est déplacé dans la zone *AlternateUserID* du descripteur d'objet avant l'émission d'un appel MQOPEN ou MQPUT1 pour la file d'attente de destination cible.

z/OS *Demandes MQI client*

Divers ID utilisateur peuvent être utilisés, en fonction des ID utilisateur et des variables d'environnement définis. Ces ID utilisateur sont vérifiés par rapport à divers profils, en fonction de l'option PUTAUT utilisée et de la spécification ou non d'un autre ID utilisateur.

Cette section décrit les ID utilisateur vérifiés pour les demandes MQI client émises via les canaux de connexion serveur pour TCP/IP et LU 6.2. L'ID utilisateur MCA et l'ID utilisateur de canal sont les mêmes que pour les canaux TCP/IP et LU 6.2 décrits dans les sections précédentes.

Pour les canaux de connexion serveur, l'ID utilisateur reçu du client est utilisé si l'attribut MCAUSER est vide.

Pour plus d'informations, voir «Contrôle d'accès pour les clients», à la page 99.

Pour les demandes du client **MQOPEN**, **MQSUB**et **MQPUT1** , utilisez les règles suivantes pour déterminer le profil vérifié:

- Si la demande spécifie des droits d'utilisateur de remplacement, une vérification est effectuée par rapport à *hlq*.ALTERNATE.USER. Profil *userid* .
- Si la demande spécifie des droits de contexte, une vérification est effectuée par rapport à *hlq*.CONTEXTE. Profil *queuename* .
- Pour toutes les demandes **MQOPEN**, **MQSUB**et **MQPUT1** , une vérification est effectuée par rapport au profil *hlq.resourcename* .

Une fois que vous avez déterminé quels profils sont vérifiés, utilisez le tableau suivant pour déterminer quels ID utilisateur sont vérifiés par rapport à ces profils.

Tableau 63. ID utilisateur vérifiés par rapport au nom de profil pour les canaux de connexion serveur de LU 6.2 et TCP/IP

Option PUTAUT indiquée sur le canal de connexion serveur	ID utilisateur de remplacement spécifié à l'ouverture?	hlq.ALTERNATE.USER.userid	Profil hlq.CONTEXT.queuename	Profil hlq.resourcename
DEF, 1 vérification	Non	-	CHL	CHL
DEF, 1 vérification	Oui	CHL	CHL	CHL
DEF, 2 vérifications	Non	-	CHL + MCA	CHL + MCA
DEF, 2 vérifications	Oui	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 vérification	Non	-	MCA	MCA
ONLYMCA, 1 vérification	Oui	MCA	MCA	MCA
ONLYMCA, 2 vérifications	Non	-	MCA	MCA
ONLYMCA, 2 vérifications	Oui	MCA	MCA	MCA + ALT

Clé :

MCA (ID utilisateur MCA)
ID utilisateur indiqué pour l'attribut de canal MCAUSER au niveau de la connexion serveur ; s'il est vide, l'ID utilisateur de l'espace adresse de l'initiateur de canal est utilisé.

CHL (ID utilisateur de canal)
Sur TCP/IP, la sécurité n'est pas prise en charge par le système de communication du canal.
Si Transport Layer Security (TLS) est utilisé et qu'un certificat numérique a été transmis par le partenaire, l'ID utilisateur associé à ce certificat (s'il est installé) ou l'ID utilisateur associé à un filtre correspondant trouvé à l'aide de RACF Certificate Name Filtering (CNF) est utilisé. Si aucun ID utilisateur associé n'est trouvé ou si TLS n'est pas utilisé, l'ID utilisateur de l'espace adresse de l'initiateur de canal est utilisé comme ID utilisateur de canal sur les canaux définis avec le paramètre PUTAUT défini sur DEF ou CTX.

Remarque : L'utilisation de RACF Certificate Name Filtering (CNF) vous permet d'affecter le même ID utilisateur RACF à plusieurs utilisateurs distants, par exemple tous les utilisateurs de la même unité organisationnelle, qui possèdent naturellement tous les mêmes droits de sécurité. Cela signifie que le serveur n'a pas besoin de disposer d'une copie du certificat de chaque utilisateur distant possible à travers le monde, et simplifie considérablement la gestion et la distribution des certificats.

Si le paramètre PUTAUT est défini sur ONLYMCA ou ALTMCA pour le canal, l'ID utilisateur du canal est ignoré et l'ID utilisateur MCA du canal de connexion serveur est utilisé. Cela s'applique également aux canaux TCP/IP utilisant TLS.

ALT (ID utilisateur de remplacement)
ID utilisateur provenant des informations de contexte (c'est-à-dire la zone *UserIdentifier*) dans le descripteur de message du message. Cet ID utilisateur est déplacé dans la zone *AlternateUserID* du descripteur d'objet ou d'abonnement avant qu'un appel **MQOPEN**, **MQSUB** ou **MQPUT1** ne soit émis pour le compte de l'application client.

z/OS *Exemple d'initiateur de canal*
Exemple de vérification des ID utilisateur par rapport aux profils RACF .

Un utilisateur effectue une opération **MQPUT1** sur une file d'attente du gestionnaire de files d'attente QM01 qui se résout en une file d'attente appelée QB sur le gestionnaire de files d'attente QM02. Le message est envoyé sur un canal TCP/IP appelé QM01.TO.QM02. RESLEVEL est défini sur NONE et l'ouverture est effectuée avec un autre ID utilisateur et une vérification de contexte. La définition de canal récepteur comporte PUTAUT (CTX) et l'ID utilisateur MCA est défini. Quels ID utilisateur sont utilisés sur le canal récepteur pour placer le message dans la file d'attente QB?

Réponse: Tableau 55, à la page 242 indique que deux ID utilisateur sont vérifiés car RESLEVEL est défini sur NONE.

Tableau 61, à la page 248 montre que, avec PUTAUT défini sur CTX et 2 vérifications, les ID utilisateur suivants sont vérifiés:

- L'ID utilisateur de l'initiateur de canal et l'ID utilisateur MCAUSER sont vérifiés par rapport à hlq.ALTERNATE.USER.userid .
- L'ID utilisateur de l'initiateur de canal et l'ID utilisateur MCAUSER sont vérifiés par rapport au profil hlq.CONTEXT.queuename .
- L'ID utilisateur de l'initiateur de canal et l'ID utilisateur de remplacement spécifiés dans le descripteur de message (MQMD) sont vérifiés par rapport au profil hlq.Q2 .

z/OS *ID utilisateur utilisés par l'agent de mise en file d'attente intra-groupe*
Les ID utilisateur vérifiés lorsque l'agent de mise en file d'attente intra-groupe ouvre des files d'attente de destination sont déterminés par les valeurs des attributs de gestionnaire de files d'attente IGQAUT et IGQUSER.

Les ID utilisateur possibles sont les suivants:

ID utilisateur de mise en file d'attente intra-groupe (IGQ)
ID utilisateur déterminé par l'attribut IGQUSER du gestionnaire de files d'attente de réception. Si ce paramètre est mis à blanc, l'ID utilisateur du gestionnaire de files d'attente de réception est utilisé. Toutefois, comme le gestionnaire de files d'attente de réception est autorisé à accéder à toutes les files d'attente qui lui sont définies, les contrôles de sécurité ne sont pas effectués pour l'ID utilisateur du gestionnaire de files d'attente de réception. Dans ce cas :

- Si un seul ID utilisateur doit être vérifié et que l'ID utilisateur est celui du gestionnaire de files d'attente de réception, aucun contrôle de sécurité n'est effectué. Cela peut se produire lorsque IGQAUT est défini sur ONLYIGQ ou ALTIGQ.
- Si deux ID utilisateur doivent être vérifiés et que l'un des ID utilisateur est celui du gestionnaire de files d'attente de réception, des contrôles de sécurité sont effectués pour l'autre ID utilisateur uniquement. Cela peut se produire lorsque IGQAUT est défini sur DEF, CTX ou ALTIGQ.
- Si deux ID utilisateur doivent être vérifiés et que les deux ID utilisateur sont ceux du gestionnaire de files d'attente de réception, aucun contrôle de sécurité n'est effectué. Cela peut se produire lorsque IGQAUT est défini sur ONLYIGQ.

ID utilisateur du gestionnaire de files d'attente émetteur (SND)
ID utilisateur du gestionnaire de files d'attente dans le groupe de partage de files d'attente qui a inséré le message dans SYSTEM.QSG.TRANSMIT.QUEUE.

Autre ID utilisateur (ALT)
ID utilisateur spécifié dans la zone *UserIdentifier* du descripteur de message du message.

Tableau 64. ID utilisateur vérifiés par rapport au nom de profil pour la mise en file d'attente intra-groupe

Option IGQAUT spécifiée sur le gestionnaire de files d'attente de réception	hlq.ALTERNATE.USER.userid	Profil hlq.CONTEXT.queuename	Profil hlq.resourcename
DEF, 1 vérification	-	SND	SND
DEF, 2 vérifications	-	SND + IGQ	SND + IGQ
CTX, 1 vérification	SND	SND	SND
CTX, 2 vérifications	SND + IGQ	SND + IGQ	SND + ALT
ONLYIGQ, 1 vérification	-	IGQ	IGQ
ONLYIGQ, 2 vérifications	-	IGQ	IGQ
ALTIGQ, 1 vérification	-	IGQ	IGQ
ALTIGQ, 2 vérifications	IGQ	IGQ	IGQ + ALT

Clé :

ALT
Autre ID utilisateur.

IGQ
ID utilisateur de mise en file d'attente intra-groupe.

SND
ID utilisateur du gestionnaire de files d'attente d'envoi.

z/OS *ID utilisateur et niveaux UACC vides*

Si un ID utilisateur vide se produit, un utilisateur non défini RACF est connecté. N'accordez pas un accès étendu à l'utilisateur non défini.

Des ID utilisateur vides peuvent exister lorsqu'un utilisateur manipule des messages à l'aide de la sécurité du contexte ou d'un autre utilisateur, ou lorsqu'un ID utilisateur vide est transmis à IBM MQ . Par exemple, un ID utilisateur vide est utilisé lorsqu'un message est écrit dans la file d'attente d'entrée de la commande système sans contexte.

Remarque : Un ID utilisateur de " * " (c'est-à-dire un astérisque suivi de sept espaces) est traité comme un ID utilisateur non défini.

IBM MQ transmet l'ID utilisateur vide à RACF et un utilisateur RACF non défini est connecté. Tous les contrôles de sécurité utilisent ensuite l'accès universel (UACC) pour le profil approprié. Selon la manière dont vous avez défini vos niveaux d'accès, l'UACC peut donner à l'utilisateur non défini un accès étendu.

Par exemple, si vous exécutez cette commande RACF à partir de TSO:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.EVERYONE UACC(UPDATE)
```

vous définissez un profil qui active à la fois les ID utilisateur définis par z/OS(qui n'ont pas été placés dans la liste d'accès) et l'ID utilisateur non défini RACF pour y placer des messages et en extraire des messages.

Pour vous protéger contre les ID utilisateur vides, vous devez planifier soigneusement vos niveaux d'accès et limiter le nombre de personnes pouvant utiliser le contexte et la sécurité des autres utilisateurs. Vous devez empêcher les personnes utilisant l'ID utilisateur non défini RACF d'accéder aux ressources auxquelles elles ne doivent pas accéder. Toutefois, dans le même temps, vous devez autoriser l'accès aux personnes ayant des ID utilisateur définis. Pour ce faire, vous pouvez spécifier l'ID utilisateur astérisque (*) dans une commande RACF PERMIT, en donnant accès aux ressources pour tous les ID utilisateur définis. Par conséquent, tous les ID utilisateur non définis (tels que " * ") sont refusés. Par exemple, ces commandes RACF empêchent l'ID utilisateur non défini RACF d'accéder à la file d'attente pour insérer ou extraire des messages:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY UACC(NONE)
PERMIT Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY CLASS(MQQUEUE) ACCESS(UPDATE) ID(*)
```

z/OS ID utilisateur z/OS et authentification multi-facteur (MFA)

IBM Multi-Factor Authentication for z/OS permet aux administrateurs de sécurité z/OS d'améliorer l'authentification SAF en demandant aux utilisateurs identifiés d'utiliser plusieurs facteurs d'authentification (par exemple, un mot de passe et un jeton cryptographique) pour se connecter à un système z/OS . IBM L'authentification multi-facteur prend également en charge les technologies de génération de mot de passe à utilisation unique basée sur le temps, telles que RSA SecureId.

Dans la plupart des cas, IBM MQ ne sait pas comment les utilisateurs se sont "connectés" au CICS ou aux systèmes de traitement par lots qui pilotent le travail IBM MQ , les données d'identification de l'ID utilisateur connecté sont associées à la tâche z/OS ou à l'espace adresse et IBM MQ les utilise pour vérifier l'autorisation d'accès aux ressources. Les ID utilisateur activés pour l'authentification multi-facteur peuvent être utilisés pour l'autorisation des ressources IBM MQ et l'authentification via des tickets de passage utilisés avec les ponts CICS et IMS .

Important : Toutefois, des considérations spéciales s'appliquent lors de l'utilisation d'applications, telles que IBM MQ Explorer, qui transmettent un ID utilisateur et des données d'identification par mot de passe sur un appel API MQCONNX avec l'option *MQCSP_AUTH_USER_ID_AND_PWD* . IBM MQ n'a pas la possibilité de transmettre des données d'identification supplémentaires sur cette demande d'API.

Les limitations et les solutions de contournement potentielles sont décrites dans le texte suivant.

IBM MQ Explorer

IBM MQ Explorer ne peut pas être utilisé pour se connecter à un système z/OS avec un ID utilisateur pour lequel l'authentification multi-facteur est activée car il n'existe pas de fonction permettant de transmettre un deuxième facteur d'authentification de IBM MQ Explorer à z/OS.

En outre, il existe deux mécanismes différents utilisés par IBM MQ Explorer pour réutiliser un ID utilisateur et des données d'identification par mot de passe, qui nécessitent une attention particulière lorsque des mots de passe à usage unique sont utilisés:

1. IBM MQ Explorer a la possibilité de stocker les mots de passe dans un format brouillé sur la machine locale pour la connexion ultérieure. Cette fonction doit être désactivée en demandant un mot de passe à l'explorateur chaque fois qu'une connexion est établie avec le gestionnaire de files d'attente z/OS .

   Pour ce faire, procédez comme suit :

   a. Sélectionnez **Gestionnaires de files d'attente**.
   b. Dans la liste qui s'affiche, sélectionnez le gestionnaire de files d'attente dont vous avez besoin et cliquez avec le bouton droit de la souris sur ce gestionnaire de files d'attente.
   c. Sélectionnez **Détails de la connexion** dans la liste de menus qui s'affiche.
   d. Sélectionnez **Propriétés** dans la liste de menu suivante et sélectionnez l'onglet **ID utilisateur** .

      Veillez à sélectionner le bouton d'option **prompt for password** .
2. Diverses opérations dans IBM MQ Explorer, telles que l'exploration des messages dans les files d'attente, le test des abonnements, etc., démarrent une nouvelle unité d'exécution qui s'authentifie

auprès de IBM MQ à l'aide des données d'identification utilisées lors de la connexion. Etant donné que les données d'identification par mot de passe ne peuvent pas être réutilisées, vous ne pouvez pas utiliser ces opérations.

Il existe deux solutions de contournement possibles au niveau de la configuration de l'authentification multi-facteur pour ces problèmes:

- Utilisez l'exclusion d'ID application de l'authentification multi-facteur pour exclure complètement les tâches IBM MQ du traitement de l'authentification multi-facteur.

  Pour ce faire, exécutez les commandes suivantes:

  1. `RDEFINE MFADEF MFABYPASS.USERID.chinuser`

     où *chinuser* est l'ID utilisateur de niveau d'espace adresse de l'initiateur de canal (associé à l'initiateur de canal via la classe STC)

  2. `PERMIT MFABYPASS.USERID.chinuser CLASS MFADEF ACCESS(READ) ID(explorer user)`

  Pour plus d'informations sur cette approche, voir Contournement de l'authentification multi-facteur IBM pour les applications.
- Utilisation de la prise en charge externe sur l'authentification multi-facteur, qui a été introduite avec l' IBM authentification multi-facteur 1.2. Avec cette approche, vous vous préauthentifiez auprès du serveur Web IBM MFA et, en plus de votre ID utilisateur et de votre mot de passe, vous spécifiez une authentification supplémentaire déterminée par la règle. Le serveur IBM MFA génère des données d'identification de jeton de cache que vous spécifiez ensuite dans la boîte de dialogue d'authentification IBM MQ Explorer . L'administrateur de la sécurité peut autoriser la réexécution de ces données d'identification pendant une période de temps raisonnable, ce qui permet une utilisation normale de IBM MQ Explorer .

Pour plus d'informations sur cette approche, voir Introduction to IBM MFA.

z/OS IBM MQ for z/OSGestion de la sécurité

IBM MQ utilise une table en mémoire pour stocker les informations relatives à chaque utilisateur et les demandes d'accès effectuées par chaque utilisateur. Pour gérer cette table de manière efficace et réduire le nombre de demandes émises par IBM MQ au gestionnaire de sécurité externe (ESM), un certain nombre de contrôles sont disponibles.

Ces contrôles sont disponibles via les panneaux d'opérations et de contrôle et les commandes IBM MQ .

z/OS *Revérification de l'ID utilisateur*

Si la définition RACF d'un utilisateur qui utilise des ressources IBM MQ a été modifiée, par exemple en connectant l'utilisateur à un nouveau groupe, vous pouvez demander au gestionnaire de files d'attente de reconnecter cet utilisateur lors de sa prochaine tentative d'accès à une ressource IBM MQ . Pour ce faire, utilisez la commande IBM MQ RVERIFY SECURITY.

- L'utilisateur HX0804 obtient et place les messages dans les files d'attente PAYROLL sur le gestionnaire de files d'attente PRD1. Toutefois, HX0804 requiert désormais l'accès à certaines des files d'attente de pensions du même gestionnaire de files d'attente (PRD1).
- L'administrateur de la sécurité des données connecte l'utilisateur HX0804 au groupe RACF qui permet d'accéder aux files d'attente de pensions.
- Pour que HX0804 puisse accéder aux files d'attente de pensions immédiatement (c'est-à-dire sans arrêter le gestionnaire de files d'attente PRD1 ou attendre que HX0804 ait été arrêté), vous devez utiliser la commande IBM MQ :

```
RVERIFY SECURITY(HX0804)
```

Remarque : Si vous désactivez le délai d'attente de l'ID utilisateur pendant de longues périodes (jours ou même des semaines) pendant que le gestionnaire de files d'attente est en cours d'exécution, vous devez

vous rappeler d'exécuter la commande RVERIFY SECURITY pour tous les utilisateurs qui ont été révoqués ou supprimés pendant cette période.

z/OS *Délais d'attente de l'ID utilisateur*

Vous pouvez demander à IBM MQ de déconnecter un utilisateur d'un gestionnaire de files d'attente après une période d'inactivité.

Lorsqu'un utilisateur accède à une ressource IBM MQ , le gestionnaire de files d'attente tente de le connecter au gestionnaire de files d'attente (si la sécurité du sous-système est active). Cela signifie que l'utilisateur est authentifié auprès du gestionnaire ESM. Cet utilisateur reste connecté à IBM MQ jusqu'à ce que le gestionnaire de files d'attente soit arrêté ou que l'ID utilisateur soit *arrivé à expiration* (l'authentification est périmée) ou revérifié (reauthentifié).

Lorsqu'un utilisateur arrive à expiration, l'ID utilisateur est *déconnecté* dans le gestionnaire de files d'attente et toutes les informations liées à la sécurité conservées pour cet utilisateur sont supprimées. La connexion et la déconnexion de l'utilisateur dans le gestionnaire de files d'attente ne sont pas apparentes pour le programme d'application ou pour l'utilisateur.

Les utilisateurs peuvent bénéficier d'un délai d'attente lorsqu'ils n'ont pas utilisé de ressources IBM MQ pendant une durée prédéterminée. Cette période est définie par la commande MQSC ALTER SECURITY.

Deux valeurs peuvent être spécifiées dans la commande ALTER SECURITY:

TIMEOUT
Période, en minutes, pendant laquelle un ID utilisateur inutilisé et ses ressources associées peuvent rester dans le gestionnaire de files d'attente IBM MQ .

INTERVAL
Délai en minutes entre les vérifications des ID utilisateur et des ressources associées, afin de déterminer si le *TIMEOUT* a expiré.

Par exemple, si la valeur *TIMEOUT* est 30 et que la valeur *INTERVAL* est 10, toutes les 10 minutes, IBM MQ vérifie les ID utilisateur et leurs ressources associées afin de déterminer si elles n'ont pas été utilisées pendant 30 minutes. En cas de détection d'un ID utilisateur expiré, cet ID est déconnecté du gestionnaire de files d'attente. Si des informations de ressource expirées associées à des ID utilisateur non expirées sont trouvées, ces informations de ressource sont supprimées. Si vous ne souhaitez pas définir de délai d'attente pour les ID utilisateur, définissez la valeur *INTERVAL* sur zéro. Toutefois, si la valeur *INTERVAL* est égale à zéro, la mémoire occupée par les ID utilisateur et les ressources associées n'est pas libérée tant que vous n'émettez pas une commande **REFRESH SECURITY** ou **RVERIFY SECURITY** .

L'optimisation de cette valeur peut être importante si vous disposez de plusieurs utilisateurs. Si vous définissez de petites valeurs d'intervalle et de délai d'attente, les ressources qui ne sont plus nécessaires sont libérées.

Remarque : Si vous utilisez des valeurs pour *INTERVAL* ou *TIMEOUT* autres que les valeurs par défaut, vous devez entrer à nouveau la commande à chaque démarrage du gestionnaire de files d'attente. Pour ce faire, vous pouvez placer la commande **ALTER SECURITY** dans le fichier CSQINP1 de ce gestionnaire de files d'attente.

z/OS *Régénération de la sécurité du gestionnaire de files d'attente sous z/OS*

IBM MQ for z/OS met en cache RACF les données pour améliorer les performances. Lorsque vous modifiez certaines classes de sécurité, vous devez actualiser ces informations mises en cache. Actualisez la sécurité de manière peu fréquente, pour des raisons de performances. Vous pouvez également choisir d'actualiser uniquement les informations de sécurité TLS.

Lorsqu'une file d'attente est ouverte pour la première fois (ou pour la première fois depuis une actualisation de la sécurité), IBM MQ effectue une vérification RACF pour obtenir les droits d'accès de l'utilisateur et place ces informations dans le cache. Les données mises en cache incluent les ID utilisateur et les ressources sur lesquelles la vérification de la sécurité a été effectuée. Si la file d'attente est à nouveau ouverte par le même utilisateur, la présence des données en cache signifie que IBM MQ n'a pas à émettre de vérifications RACF , ce qui améliore les performances. L'action d'une actualisation de la sécurité consiste à supprimer les informations de sécurité mises en cache et à forcer IBM MQ

à effectuer une nouvelle vérification par rapport à RACF. Chaque fois que vous ajoutez, modifiez ou supprimez un profil de ressource RACF qui se trouve dans la classe MQADMIN, MXADMIN, MQPROC, MXPROC, MQQUEUE, MXQUEUE, MQNLIST, MXNLIST ou MXTOPIC, vous devez indiquer aux gestionnaires de files d'attente qui utilisent cette classe d'actualiser les informations de sécurité qu'ils détiennent. Pour ce faire, exécutez les commandes suivantes:

- La commande RACF SETROPTS RACLIST (classname) REFRESH pour effectuer une actualisation au niveau RACF .
- La commande IBM MQ REFRESH SECURITY pour actualiser les informations de sécurité détenues par le gestionnaire de files d'attente. Cette commande doit être émise par chaque gestionnaire de files d'attente qui accède aux profils modifiés. Si vous disposez d'un groupe de partage de files d'attente, vous pouvez utiliser l'attribut de portée de commande pour diriger la commande vers tous les gestionnaires de files d'attente du groupe.

  Remarque : Si vous avez connecté un nouvel utilisateur à un groupe existant, vous devez exécuter la commande IBM MQ RVERIFY SECURITY(userid). La commande REFRESH SECURITY (*) ne permet pas au gestionnaire de files d'attente de reconnecter cet utilisateur, la prochaine fois qu'il tente d'accéder à une ressource IBM MQ .

Si vous utilisez des profils génériques dans l'une des classes IBM MQ , vous devez également exécuter des commandes d'actualisation RACF normales si vous modifiez, ajoutez ou supprimez des profils génériques. Par exemple, SETROPTS GENERIC (classname) ACTUALISER.

Toutefois, si un profil de ressource RACF est ajouté, modifié ou supprimé et que la ressource à laquelle il s'applique n'a pas encore été consultée (aucune information n'est donc mise en cache), IBM MQ utilise les nouvelles informations RACF sans qu'une commande REFRESH SECURITY soit émise.

Si l'audit RACF est activé (par exemple, à l'aide de la commande RACF RALTER AUDIT (access-Tentative (audit_access_level))), aucune mise en cache n'a lieu, et par conséquent, IBM MQ fait directement référence à l'espace de données RACF pour chaque vérification. Les modifications sont donc prises en compte immédiatement et REFRESH SECURITY n'est pas nécessaire pour accéder aux modifications. Vous pouvez vérifier si l'audit RACF est exécuté à l'aide de la commande RACF RLIST. Par exemple, vous pouvez émettre la commande

```
RLIST MQQUEUE (qmgr.SYSTEM.COMMAND.INPUT) GEN
```

et recevoir les résultats

```
CLASS      NAME
-----      ----
MQQUEUE    QP*.SYSTEM.COMMAND.*.** (G)
    AUDITING
    --------
    FAILURES(READ)
```

Indique que l'audit est défini sur. Pour plus d'informations, reportez-vous au manuel *z/OS Security Server RACF Auditor's Guide* et au manuel *z/OS Security Server RACF Command Language Reference*.

Le Figure 17, à la page 258 récapitule les situations dans lesquelles les informations de sécurité sont mises en cache et dans lesquelles les informations mises en cache sont utilisées.

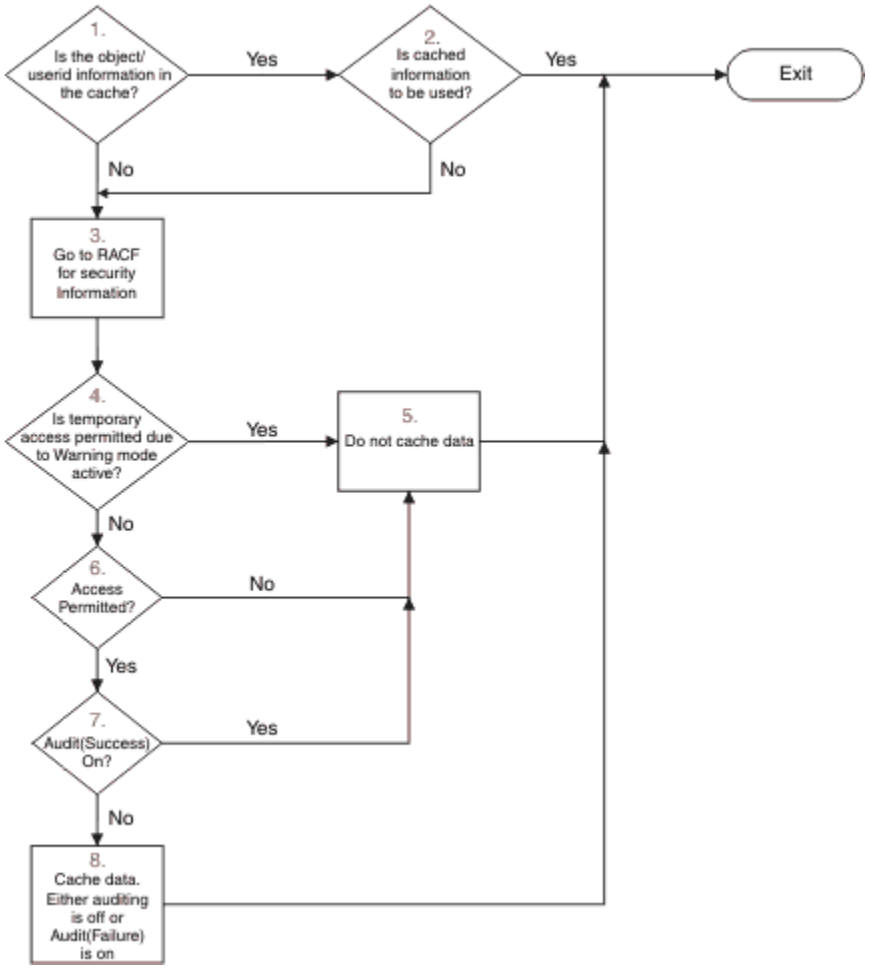


Figure 17. Flux logique pour la mise en cache de la sécurité IBM MQ

Si vous modifiez vos paramètres de sécurité en ajoutant ou en supprimant des profils de commutation dans les classes MQADMIN ou MXADMIN, utilisez l'une des commandes suivantes pour prendre en compte ces modifications de manière dynamique:

ACTUALISER LA SECURITE (*)
ACTUALISER LA SECURITE (MQADMIN)
ACTUALISER LA SECURITE (MXADMIN)

Cela signifie que vous pouvez activer de nouveaux types de sécurité ou les désactiver sans avoir à redémarrer le gestionnaire de files d'attente.

Pour des raisons de performances, il s'agit des seules classes affectées par la commande REFRESH SECURITY. Vous n'avez pas besoin d'utiliser REFRESH SECURITY si vous modifiez un profil dans les classes MQCONN ou MQCMDS.

Remarque : Une actualisation de la classe MQADMIN ou MXADMIN n'est pas requise si vous modifiez un profil de sécurité RESLEVEL.

Pour des raisons de performances, utilisez REFRESH SECURITY aussi rarement que possible, idéalement aux heures creuses. Vous pouvez réduire le nombre d'actualisations de sécurité en connectant des utilisateurs à des groupes RACF qui figurent déjà dans la liste d'accès pour les profils IBM MQ , au lieu de placer des utilisateurs individuels dans les listes d'accès. De cette manière, vous modifiez l'utilisateur plutôt que le profil de ressource. Vous pouvez également RVERIFY SECURITY à l'utilisateur approprié au lieu d'actualiser la sécurité.

A titre d'exemple de REFRESH SECURITY, supposons que vous définissiez les nouveaux profils pour protéger l'accès aux files d'attente commençant par INSURANCE.LIFE sur le gestionnaire de files d'attente PRMQ. Vous utilisez les commandes RACF suivantes:

```
RDEFINE MQQUEUE PRMQ.INSURANCE.LIFE.** UACC(NONE)
PERMIT PRMQ.INSURANCE.LIFE.** ID(LIFEGRP) ACCESS(UPDATE)
```

Vous devez exécuter la commande suivante pour indiquer à RACF d'actualiser les informations de sécurité qu'elle contient, par exemple:

```
SETROPTS RACLIST(MQQUEUE) REFRESH
```

Etant donné que ces profils sont génériques, vous devez indiquer à RACF d'actualiser les profils génériques pour MQQUEUE. Exemple :

```
SETROPTS GENERIC(MQQUEUE) REFRESH
```

Vous devez ensuite utiliser cette commande pour indiquer au gestionnaire de files d'attente PRMQ que les profils de file d'attente ont été modifiés:

```
REFRESH SECURITY(MQQUEUE)
```

Actualisation de la sécurité SSL/TLS

Pour actualiser la vue mise en cache du référentiel de clés TLS, exécutez la commande REFRESH SECURITY avec l'option TYPE (SSL). Cela vous permet de mettre à jour certains de vos paramètres TLS sans avoir à redémarrer votre initiateur de canal.

z/OS *Affichage du statut de sécurité*

Pour afficher le statut des commutateurs de sécurité et d'autres contrôles de sécurité, exécutez la commande MQSC DISPLAY SECURITY.

La figure suivante illustre la sortie typique de la commande DISPLAY SECURITY ALL.

```
CSQH015I +CSQ1 Security timeout = 54 MINUTES
CSQH016I +CSQ1 Security interval = 12 MINUTES
CSQH030I +CSQ1 Security switches ...
CSQH034I +CSQ1 SUBSYSTEM: ON, 'SQ05.NO.SUBSYS.SECURITY' not found
CSQH032I +CSQ1 QMGR: ON, 'CSQ1.YES.QMGR.CHECKS' found
CSQH031I +CSQ1 QSG: OFF, 'SQ05.NO.QSG.CHECKS' found
CSQH031I +CSQ1 CONNECTION: OFF, 'CSQ1.NO.CONNECT.CHECKS' found
CSQH034I +CSQ1 COMMAND: ON, 'CSQ1.NO.COMMAND.CHECKS' not found
CSQH031I +CSQ1 CONTEXT: OFF, 'CSQ1.NO.CONTEXT.CHECKS' found
CSQH034I +CSQ1 ALTERNATE USER: ON, 'CSQ1.NO.ALTERNATE.USER.CHECKS' not found
CSQH034I +CSQ1 PROCESS: ON, 'CSQ1.NO.PROCESS.CHECKS' not found
CSQH034I +CSQ1 NAMELIST: ON, 'CSQ1.NO.NLIST.CHECKS' not found
CSQH034I +CSQ1 QUEUE: ON, 'CSQ1.NO.QUEUE.CHECKS' not found
CSQH034I +CSQ1 TOPIC: ON, 'CSQ1.NO.TOPIC.CHECKS' not found
CSQH031I +CSQ1 COMMAND RESOURCES: OFF, 'CSQ1.NO.CMD.RESC.CHECKS' found
CSQ9022I +CSQ1 CSQHPDTC ' DISPLAY SECURITY' NORMAL COMPLETION
```

Figure 18. Sortie standard de la commande DISPLAY SECURITY

L'exemple montre que le gestionnaire de files d'attente qui a répondu à la commande possède un sous-système, une commande, un autre utilisateur, un processus, une liste de noms et une sécurité de file d'attente actifs au niveau du gestionnaire de files d'attente mais pas au niveau du groupe de partage de files d'attente. La connexion, la ressource de commande et la sécurité de contexte ne sont pas actives. Il indique également que les délais d'attente de l'ID utilisateur sont actifs et que, toutes les 12 minutes, le

gestionnaire de files d'attente recherche les ID utilisateur qui n'ont pas été utilisés dans ce gestionnaire de files d'attente pendant 54 minutes et les supprime.

Remarque : Cette commande affiche le statut de sécurité en cours. Il ne reflète pas nécessairement le statut en cours des profils de commutation définis sur RACF, ni le statut des classes RACF . Par exemple, les profils de commutation peuvent avoir été modifiés depuis le dernier redémarrage de ce gestionnaire de files d'attente ou de la commande REFRESH SECURITY.

z/OS Tâches d'installation de la sécurité pour z/OS

Après avoir installé et personnalisé IBM MQ, autorisez les procédures de tâche démarrée à RACF, autorisez l'accès à diverses ressources et configurez des définitions RACF . Vous pouvez éventuellement configurer votre système pour TLS.

Lorsque IBM MQ est installé et personnalisé pour la première fois, vous devez effectuer les tâches liées à la sécurité suivantes:

1. Configurez la sécurité de l'ensemble de données et du système IBM MQ en:
   - Autorisation de la procédure de tâche démarrée du gestionnaire de files d'attente xxxxMSTR et de la procédure de tâche démarrée de la mise en file d'attente répartie xxxxCHIN à exécuter sous RACF.
   - Autorisation d'accès aux fichiers du gestionnaire de files d'attente.
   - Autorisation d'accès aux ressources pour les ID utilisateur qui utiliseront le gestionnaire de files d'attente et les programmes utilitaires.
   - Autorisation d'accès pour les gestionnaires de files d'attente qui utiliseront les structures de liste de l'unité de couplage.
   - Autorisation d'accès pour les gestionnaires de files d'attente qui utiliseront Db2.
2. Configurez les définitions RACF pour la sécurité IBM MQ .
3. Si vous souhaitez utiliser le protocole TLS (Transport Layer Security), préparez votre système à utiliser des certificats et des clés.

z/OS *Configuration de la sécurité des fichiers IBM MQ for z/OS*

Il existe de nombreux types d'utilisateur IBM MQ . Utilisez RACF pour contrôler leur accès aux fichiers système.

Les utilisateurs possibles des ensembles de données IBM MQ incluent les entités suivantes:

- Le gestionnaire de files d'attente lui-même.
- Initiateur de canal
- Les administrateurs IBM MQ , qui doivent créer des fichiers IBM MQ , exécuter des programmes utilitaires et des tâches similaires.
- Les programmeurs d'application qui doivent utiliser les fichiers de stockage fournis par IBM MQincluent des jeux de données, des macros et des ressources similaires.
- Applications impliquant une ou plusieurs des fonctions suivantes:
  - Travaux par lots
  - Utilisateurs TSO
  - Régions CICS
  - Régions IMS
- Fichiers CSQOUTX et CSQSNAP
- Files d'attente dynamiques SYSTEM.CSQXCMD.*

Pour tous ces utilisateurs potentiels, protégez les fichiers IBM MQ avec RACF.

Vous devez également contrôler l'accès à tous vos fichiers'CSQINP'.

z/OS *Autorisation RACF des procédures de tâche démarrée*

Certains fichiers IBM MQ sont destinés à l'utilisation exclusive du gestionnaire de files d'attente. Si vous protégez vos fichiers IBM MQ à l'aide de RACF, vous devez également autoriser la procédure de tâche démarrée du gestionnaire de files d'attente xxxxMSTRet la procédure de tâche démarrée de mise en file d'attente répartie xxxxCHIN, à l'aide de RACF. Pour ce faire, utilisez la classe STARTED. Vous pouvez également utiliser la table des procédures démarrées (ICHRIN03), mais vous devez ensuite effectuer un IPL de votre système z/OS pour que les modifications soient prises en compte.

Pour plus d'informations, voir le manuel *z/OS Security Server RACF System Programmer's Guide*.

L'ID utilisateur RACF identifié doit disposer des droits d'accès requis aux fichiers dans la procédure de tâche démarrée. Par exemple, si vous associez une procédure de tâche démarrée de gestionnaire de files d'attente appelée CSQ1MSTR à l'ID utilisateur RACF QMGRCSQ1, l'ID utilisateur QMGRCSQ1 doit avoir accès aux ressources z/OS auxquelles accède le gestionnaire de files d'attente CSQ1 .

De plus, le contenu de la zone GROUP dans l'ID utilisateur du gestionnaire de files d'attente doit être identique au contenu de la zone GROUP dans le profil STARTED de ce gestionnaire de files d'attente. Si le contenu de chaque zone GROUP ne correspond pas, l'ID utilisateur approprié ne peut pas entrer dans le système. Cette situation entraîne l'exécution de IBM MQ avec un ID utilisateur non défini et, par conséquent, sa fermeture en raison d'une violation de sécurité.

L'attribut TRUSTED ne doit pas être défini pour les ID utilisateur RACF associés aux procédures de tâche démarrée du gestionnaire de files d'attente et de l'initiateur de canal.

z/OS *Autorisation de l'accès aux fichiers*

Les fichiers IBM MQ doivent être protégés de sorte qu'aucun utilisateur non autorisé ne puisse exécuter une instance de gestionnaire de files d'attente ou accéder à des données de gestionnaire de files d'attente. Pour ce faire, utilisez la protection de fichier z/OS RACF normale.

Le Tableau 65, à la page 261 récapitule l'accès RACF que la procédure de tâche démarrée du gestionnaire de files d'attente doit avoir aux différents fichiers.

Tableau 65. Accès RACF aux fichiers associés à un gestionnaire de files d'attente

RACFaccès	**Les ensembles de données**
READ	• `thlqual.SCSQAUTH` et `thlqual.SCSQANLx` (où x est la lettre de langue de votre langue nationale). • Les fichiers référencés par CSQINP1, CSQINP2 et CSQXLIB dans la procédure de tâche démarrée du gestionnaire de files d'attente. • Fichiers SMDS appartenant à d'autres gestionnaires de files d'attente du groupe. • Fichiers journaux, fichiers d'amorce et fichiers journaux d'archivage pour les autres gestionnaires de files d'attente du groupe.
UPDATE	• Tous les ensembles de pages et les fichiers journaux et d'amorce. • Fichiers SMDS appartenant à un gestionnaire de files d'attente
ALTER	• Tous les fichiers journaux d'archivage.

Le Tableau 66, à la page 262 récapitule l'accès RACF que la procédure de tâche démarrée pour la mise en file d'attente répartie doit avoir aux différents fichiers.

Tableau 66. Accès RACF aux fichiers associés à la mise en file d'attente répartie	
RACFaccès	**Les ensembles de données**
READ	• thlqual.SCSQAUTH, thlqual.SCSQANLx (où x est la lettre de langue de votre langue nationale) et thlqual.SCSQMVR1. • Fichiers de la bibliothèque LE. • Fichiers référencés par CSQXLIB et CSQINPX dans la procédure de tâche démarrée de l'initiateur de canal.
UPDATE	• Fichiers CSQOUTX et CSQSNAP

Pour plus d'informations, voir le manuel *z/OS Security Server RACF Security Administrator's Guide*.

V 9.1.4 z/OS *Chiffrement des fichiers*

Les ensembles de données IBM MQ peuvent être chiffrés avec le chiffrement des ensembles de données z/OS , de sorte que les données soient protégées ou pour des raisons réglementaires.

Vous pouvez protéger tous les ensembles de pages, les journaux actifs, les journaux d'archivage et les fichiers d'amorçage (BSDS) avec le chiffrement des fichiers z/OS .

Avertissement : Vous ne pouvez pas protéger les fichiers de messages partagés (SMDS) avec le chiffrement des fichiers z/OS par IBM MQ for z/OS 9.1.3 ou version antérieure.

Voir la section Confidentiality for data at rest on IBM MQ for z/OS with data set encryption pour plus d'informations.

z/OS *Configuration de la sécurité des ressources IBM MQ for z/OS*

Il existe de nombreux types d'utilisateur IBM MQ . Utilisez RACF pour contrôler leur accès aux ressources IBM MQ .

Les utilisateurs possibles des ressources IBM MQ , telles que les files d'attente et les canaux, incluent les entités suivantes:

- Le gestionnaire de files d'attente lui-même.
- Initiateur de canal
- Les administrateurs IBM MQ , qui doivent créer des fichiers IBM MQ , exécuter des programmes utilitaires et des tâches similaires
- Les programmeurs d'application qui doivent utiliser les fichiers de stockage fournis par IBM MQincluent des jeux de données, des macros et des ressources similaires.
- Applications impliquant une ou plusieurs des fonctions suivantes:
  - Travaux par lots
  - Utilisateurs TSO
  - Régions CICS
  - Régions IMS
- Fichiers CSQOUTX et CSQSNAP
- Files d'attente dynamiques SYSTEM.CSQXCMD.*

Pour tous ces utilisateurs potentiels, protégez les ressources IBM MQ avec RACF. En particulier, notez que l'initiateur de canal doit accéder à diverses ressources, comme décrit dans «Remarques relatives à la sécurité de l'initiateur de canal sous z/OS», à la page 269, et que l'ID utilisateur sous lequel il s'exécute doit donc être autorisé à accéder à ces ressources.

Si vous utilisez un groupe de partage de files d'attente, le gestionnaire de files d'attente peut émettre diverses commandes en interne, de sorte que l'ID utilisateur qu'il utilise doit être autorisé à émettre de telles commandes. Les commandes sont les suivantes :

- DEFINE, ALTER et DELETE pour chaque objet doté de QSGDISP (GROUP)
- START et STOP CHANNEL pour chaque canal utilisé avec CHLDISP (SHARED)

z/OS *Configuration de votre système z/OS pour l'utilisation de TLS*

Utilisez cette rubrique comme exemple de configuration de IBM MQ for z/OS avec TLS (Transport Layer Security) à l'aide des commandes RACF .

Si vous souhaitez utiliser TLS pour la sécurité des canaux, vous devez effectuer un certain nombre de tâches sur votre système. (Pour plus de détails sur l'utilisation des commandes RACF pour les certificats et les référentiels de clés (fichiers de clés), voir Utilisation de TLS sur z/OS .)

1. Créez un fichier de clés dans RACF pour stocker toutes les clés et tous les certificats de votre système, à l'aide de la commande RACF RACDCERT. Exemple :

   ```
   RACDCERT ID(CHINUSER) ADDRING(QM1RING)
   ```

   L'ID doit être soit l'ID utilisateur de l'espace adresse de l'initiateur de canal, soit l'ID utilisateur dont vous souhaitez être propriétaire du fichier de clés s'il doit s'agir d'un fichier de clés partagé.

2. Créez un certificat numérique pour chaque gestionnaire de files d'attente à l'aide de la commande RACF RACDCERT.

   Le libellé du certificat doit être soit la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente ajouté. Pour plus de détails voir Labels de certificat numérique. Dans cet exemple, il s'agit de `ibmWebSphereMQQM1`.

   Exemple :

   ```
   RACDCERT ID(USERID) GENCERT
   SUBJECTSDN(CN('username') O('IBM') OU('departmentname') C('England'))
   WITHLABEL('ibmWebSphereMQQM1')
   ```

3. Connectez le certificat dans RACF au fichier de clés à l'aide de la commande RACF RACDCERT. Exemple :

   ```
   RACDCERT CONNECT(ID(USERID) LABEL('ibmWebSphereMQQM1') RING(QM1RING))
   CONNECT ID(CHINUSER)
   ```

   Vous devez également connecter tous les certificats de signataire pertinents (d'une autorité de certification) au fichier de clés. Autrement dit, toutes les autorités de certification pour le certificat TLS de ce gestionnaire de files d'attente et toutes les autorités de certification pour tous les certificats TLS avec lesquels ce gestionnaire de files d'attente communique. Exemple :

   ```
   RACDCERT ID(CHINUSER)
   CONNECT(CERTAUTH LABEL('My CA') RING(QM1RING) USAGE(CERTAUTH))
   ```

4. Sur chacun de vos gestionnaires de files d'attente, utilisez la commande IBM MQ ALTER QMGR pour spécifier le référentiel de clés vers lequel le gestionnaire de files d'attente doit pointer. Par exemple, si le fichier de clés appartient à l'espace adresse de l'initiateur de canal:

```
ALTER QMGR SSLKEYR(QM1RING)
```

ou si vous utilisez un fichier de clés partagé:

```
ALTER QMGR SSLKEYR(userid/QM1RING)
```

où *userid* est l'ID utilisateur qui possède le fichier de clés partagé.

5. Les listes de révocation de certificats (CRL) permettent aux autorités de certification de révoquer des certificats qui ne sont plus dignes de confiance. Les listes de révocation de certificat sont stockées dans les serveurs LDAP. Pour accéder à cette liste sur le serveur LDAP, vous devez d'abord créer un objet AUTHINFO de AUTHTYPE CRLLDAP, à l'aide de la commande IBM MQ DEFINE AUTHINFO. Exemple :

```
DEFINE AUTHINFO(LDAP1)
AUTHTYPE(CRLLDAP)
CONNAME(ldap.server(389))
LDAPUSER('')
LDAPPWD('')
```

Dans cet exemple, la liste de révocation de certificat est stockée dans une zone publique du serveur LDAP, de sorte que les zones LDAPUSER et LDAPPWD ne sont pas nécessaires.

Ensuite, placez votre objet AUTHINFO dans une liste de noms, à l'aide de la commande IBM MQ DEFINE NAMELIST. Exemple :

```
DEFINE NAMELIST(LDAPNL) NAMES(LDAP1)
```

Enfin, associez la liste de noms à chaque gestionnaire de files d'attente à l'aide de la commande IBM MQ ALTER QMGR. Exemple :

```
ALTER QMGR SSLCRLNL(LDAPNL)
```

6. Configurez votre gestionnaire de files d'attente pour exécuter des appels TLS à l'aide de la commande IBM MQ ALTER QMGR. Cela définit les sous-tâches de serveur qui traitent uniquement les appels SSL, ce qui permet aux répartiteurs normaux de continuer le traitement normalement sans être affectés par des appels SSL. Vous devez disposer d'au moins deux de ces sous-tâches. Exemple :

```
ALTER QMGR SSLTASKS(8)
```

Cette modification prend effet uniquement lorsque l'initiateur de canal est redémarré.

7. Indiquez la spécification de chiffrement à utiliser pour chaque canal, à l'aide de la commande IBM MQ DEFINE CHANNEL ou ALTER CHANNEL. Exemple :

```
ALTER CHANNEL(LDAPCHL)
CHLTYPE(SDR)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
```

Les deux extrémités du canal doivent spécifier la même spécification de chiffrement.

z/OS Gestion des enregistrements d'authentification de canal dans un groupe de partage de files d'attente

Les enregistrements d'authentification de canal s'appliquent au gestionnaire de files d'attente sur lequel ils sont créés. Ils ne sont pas partagés dans le groupe de partage de files d'attente (QSG). Par conséquent, si tous les gestionnaires de files d'attente du groupe de partage de files d'attente doivent avoir les mêmes règles, une certaine gestion doit être effectuée pour que toutes les règles soient cohérentes.

1. Ajoutez toujours l'option `CMDSCOPE(*)` à toutes les commandes `SET CHLAUTH` . La commande sera envoyée à tous les gestionnaires de files d'attente en cours d'exécution dans le groupe de partage de files d'attente
2. Utilisez la commande `DISPLAY CHLAUTH` avec l'option `CMDSCOPE(*)` , puis analysez les réponses pour voir si les enregistrements sont identiques dans tous les gestionnaires de files d'attente. Lorsqu'une incohérence est détectée, une commande `SET CHLAUTH` peut être émise contenant la même règle que `CMDSCOPE(*)` ou `CMDSCOPE(`*qmgr-name*`)`.
3. Ajoutez un membre à la concaténation CSQINP2 du gestionnaire de files d'attente (voir Commandes d'initialisation pour plus de détails) qui comporte l'ensemble complet de règles. Ils seront lus dans le cadre du processus d'initialisation du gestionnaire de files d'attente. Si la commande `SET CHLAUTH` utilise `ACTION(ADD)` , la règle ne sera ajoutée que si elle n'existait pas. L'utilisation de `ACTION(REPLACE)` remplace une règle existante si elle existe déjà ou l'ajoute si elle n'existe pas. Le même membre peut ensuite être placé dans la concaténation CSQINP2 de tous les gestionnaires de files d'attente du groupe de partage de files d'attente.
4. Utilisez l'utilitaire CSQUTIL (voir Emission de commandes pour IBM MQ (COMMAND) pour plus de détails) pour extraire les règles d'un gestionnaire de files d'attente à l'aide de l'option `MAKEDEF` ou `MAKEREP` . Réexécuter ensuite la sortie à l'aide de CSQUTIL dans le gestionnaire de files d'attente cible.

Concepts associés

Enregistrements d'authentification de canal

Pour exercer un contrôle plus précis sur les accès accordés aux systèmes en cours de connexion au niveau d'un canal, vous pouvez utiliser les enregistrements d'authentification de canal.

z/OS Remarques sur l'audit sous z/OS

Les contrôles d'audit RACF normaux sont disponibles pour effectuer un audit de sécurité d'un gestionnaire de files d'attente. IBM MQ ne collecte pas ses propres statistiques de sécurité. Les seules statistiques sont celles qui peuvent être créées par l'audit.

L'audit RACF peut être basé sur:

- ID utilisateur
- Classes de ressources
- Profils

Pour plus d'informations, voir le manuel *z/OS Security Server RACF Auditor's Guide*.

Remarque : L'audit dégrade les performances ; plus vous implémentez d'audits, plus les performances sont dégradées. Il s'agit également d'une considération pour l'utilisation de l'option RACF WARNING.

z/OS *Audit de RESLEVEL*

Utilisez le paramètre système RESAUDIT pour contrôler la production des enregistrements d'audit RESLEVEL. RACF Les enregistrements d'audit GENERAUX sont générés.

Produisez des enregistrements d'audit RESLEVEL en définissant le paramètre système RESAUDIT sur YES. Si le paramètre RESAUDIT est défini sur NO, les enregistrements d'audit ne sont pas générés. Pour plus de détails sur la définition de ce paramètre, voir Utilisation de CSQ6SYSP.

Si RESAUDIT est défini sur YES, aucun enregistrement d'audit RACF normal n'est effectué lorsque la vérification RESLEVEL est effectuée pour déterminer l'accès d'un ID utilisateur d'espace adresse au profil hlq.RESLEVEL . A la place, IBM MQ demande à RACF de créer un enregistrement d'audit GENERAL (numéro d'événement 27). Ces vérifications ne sont effectuées qu'au moment de la connexion, de sorte que le coût des performances est minimal.

Vous pouvez générer un rapport sur les enregistrements d'audit général IBM MQ à l'aide du programme d'écriture de rapport RACF (RACFRW). Vous pouvez utiliser les commandes RACFRW suivantes pour signaler l'accès RESLEVEL:

```
RACFRW
SELECT PROCESS
EVENT GENERAL
LIST
END
```

Un exemple de rapport de RACFRW, à l'exclusion des zones *Date*, *Time*et *SYSID* , est présenté dans la Figure 19, à la page 266.

```
       RACF REPORT - LISTING OF PROCESS RECORDS                                  PAGE   4
                                E
                                V  Q
                                E  U
*JOB/USER *STEP/  --TERMINAL--  N  A
   NAME    GROUP     ID    LVL  T  L

 WS21B    MQMGRP IGJZM000   0   27 0  JOBID=(WS21B 05.111 09:44:57),USERDATA=()
    TRUSTED USER                      AUTH=(NONE),REASON=(NONE)
                                SESSION=TSOLOGON,TERMINAL=IGJZM000,
                                LOGSTR='CSQH RESLEVEL CHECK PERFORMED AGAINST
PROFILE(QM66.RESLEVEL),
                                CLASS(MQADMIN), ACCESS EQUATES TO
(CONTROL)',RESULT=SUCCESS,MQADMIN
```

Figure 19. Exemple de sortie de RACFRW affichant les enregistrements d'audit général RESLEVEL

En vérifiant les données LOGSTR dans cet exemple de sortie, vous pouvez voir que l'utilisateur TSO WS21B a un accès CONTROL à QM66.RESLEVEL. Cela signifie que tous les contrôles de sécurité des ressources sont ignorés lorsque l'utilisateur WS21B accède aux ressources QM66 .

Pour plus d'informations sur l'utilisation de RACFRW, voir le manuel *z/OS Security Server RACF Auditor's Guide*.

z/OS Personnalisation de la sécurité

Si vous souhaitez modifier le mode de fonctionnement de la sécurité IBM MQ , vous devez le faire via l'exit SAF (ICHRFR00) ou via les exits de votre gestionnaire de sécurité externe.

Pour en savoir plus sur les exits RACF , voir le manuel *z/OS Security Server RACROUTE Macro Reference* .

Remarque : Etant donné que IBM MQ optimise les appels au gestionnaire ESM, les demandes RACROUTE peuvent ne pas être effectuées, par exemple, à chaque ouverture d'une file d'attente particulière par un utilisateur particulier.

z/OS Messages de violation de sécurité sur z/OS

Une violation de sécurité est indiquée par le code retour MQRC_NOT_AUTHORIZED dans un programme d'application ou par un message dans l'historique du travail.

Le code retour MQRC_NOT_AUTHORIZED peut être renvoyé à un programme d'application pour les raisons suivantes:

- Un utilisateur n'est pas autorisé à se connecter au gestionnaire de files d'attente. Dans ce cas, vous obtenez un message ICH408I dans le journal des travaux par lots / TSO, CICSou IMS .
- Une connexion utilisateur au gestionnaire de files d'attente a échoué car, par exemple, l'ID utilisateur du travail n'est pas valide ou approprié, ou l'ID utilisateur de la tâche ou l'ID utilisateur de remplacement n'est pas valide. Un ou plusieurs de ces ID utilisateur peuvent ne pas être valides car ils ont été révoqués ou supprimés. Dans ce cas, vous obtenez un message ICHxxxx et éventuellement un message IRRxxxx dans le journal des travaux du gestionnaire de files d'attente indiquant la raison de l'échec de la connexion. Exemple :

```
ICH408I USER(NOTDFND ) GROUP(        ) NAME(???                 )
  LOGON/JOB INITIATION - USER AT TERMINAL          NOT RACF-DEFINED
IRR012I  VERIFICATION FAILED. USER PROFILE NOT FOUND
```

- Un autre utilisateur a été demandé, mais l'ID utilisateur du travail ou de la tâche n'a pas accès à l'autre ID utilisateur. Pour cet échec, vous obtenez un message de violation dans l'historique du travail du gestionnaire de files d'attente approprié.
- Une option de contexte a été utilisée ou est implicite en ouvrant une file d'attente de transmission pour la sortie, mais l'ID utilisateur du travail ou, le cas échéant, l'ID utilisateur de la tâche ou de l'autre utilisateur n'a pas accès à l'option de contexte. Dans ce cas, un message de violation est inséré dans le journal des travaux du gestionnaire de files d'attente approprié.
- Un utilisateur non autorisé a tenté d'accéder à un objet de gestionnaire de files d'attente sécurisé, par exemple une file d'attente. Dans ce cas, un message ICH408I pour la violation est inséré dans le journal des travaux du gestionnaire de files d'attente approprié. Cette violation peut être due au travail ou, le cas échéant, à la tâche ou à un autre ID utilisateur.

Les messages de violation pour la sécurité des commandes et la sécurité des ressources de commandes se trouvent également dans le journal des travaux du gestionnaire de files d'attente.

Si le message de violation ICH408I indique le nom de travail du gestionnaire de files d'attente plutôt qu'un ID utilisateur, cela est généralement dû à la spécification d'un autre ID utilisateur vide. Exemple :

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
          MQS1.PAYROLL.REQUEST CL(MQQUEUE)
          INSUFFICIENT ACCESS AUTHORITY
          ACCESS INTENT(UPDATE )  ACCESS ALLOWED(NONE   )
```

Vous pouvez déterminer qui est autorisé à utiliser des ID utilisateur de remplacement vides en vérifiant la liste d'accès du profil MQADMIN hlq.ALTERNATE.USER.-BLANK-.

Un message de violation ICH408I peut également être généré par:

- Commande envoyée à la file d'attente d'entrée des commandes système sans contexte. Les programmes écrits par l'utilisateur qui écrivent dans la file d'attente d'entrée des commandes système doivent toujours utiliser une option de contexte. Pour plus d'informations, voir «Profils pour la sécurité de contexte», à la page 221.
- Lorsque le travail accédant à la ressource IBM MQ n'est associé à aucun ID utilisateur, ou lorsqu'un adaptateur IBM MQ ne peut pas extraire l'ID utilisateur de l'environnement de l'adaptateur.

Des messages de violation peuvent également être émis si vous utilisez la sécurité au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente. Il se peut que vous obteniez des

messages indiquant qu'aucun profil n'a été trouvé au niveau du gestionnaire de files d'attente, mais que l'accès soit toujours accordé en raison d'un profil au niveau du groupe de partage de files d'attente.

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
          MQS1.PAYROLL.REQUEST CL(MQQUEUE)
          PROFILE NOT FOUND - REQUIRED FOR AUTHORITY CHECKING
          ACCESS INTENT(UPDATE )  ACCESS ALLOWED(NONE   )
```

z/OS Que faire si l'accès est autorisé ou non autorisé de manière incorrecte

En plus des étapes décrites dans le manuel *z/OS Security Server RACF Security Administrator's Guide*, utilisez cette liste de contrôle si l'accès à une ressource semble être mal contrôlé.

- Les profils de commutateur sont-ils correctement définis?
  - RACF est-il actif?
  - Les classes IBM MQ RACF sont-elles installées et actives?

    Utilisez la commande RACF , SETROPTS LIST, pour vérifier cela.
  - La commande IBM MQ DISPLAY SECURITY permet d'afficher le statut en cours du commutateur à partir du gestionnaire de files d'attente.
  - Vérifiez les profils de commutateur dans la classe MQADMIN.

    Pour cela, utilisez les commandes RACF , SEARCH et RLIST.
  - Vérifiez à nouveau les profils de commutation RACF en exécutant la commande IBM MQ REFRESH SECURITY (MQADMIN).
- Le profil de ressource RACF a-t-il été modifié? Par exemple, l'accès universel au profil a-t-il été modifié ou la liste d'accès du profil a-t-elle été modifiée?
  - Le profil est-il générique?

    Si c'est le cas, exécutez la commande RACF , SETROPTS GENERIC (classname) ACTUALISER.
  - Avez-vous actualisé la sécurité sur ce gestionnaire de files d'attente?

    Si nécessaire, exécutez la commande RACF SETROPTS RACLIST (classname) ACTUALISER.

    Si nécessaire, exécutez la commande IBM MQ REFRESH SECURITY (*).
- La définition RACF de l'utilisateur a-t-elle été modifiée? Par exemple, l'utilisateur a-t-il été connecté à un nouveau groupe ou le droit d'accès de l'utilisateur a-t-il été révoqué?
  - Avez-vous revérifié l'utilisateur à l'aide de la commande IBM MQ RVERIFY SECURITY (userid)?
- Les contrôles de sécurité sont-ils ignorés en raison de RESLEVEL?
  - Vérifiez l'accès de l'ID utilisateur de connexion au profil RESLEVEL. Utilisez les enregistrements d'audit RACF pour déterminer la valeur de RESLEVEL.
  - Pour les canaux, n'oubliez pas que le niveau d'accès de l'ID utilisateur de l'initiateur de canal à RESLEVEL est hérité par tous les canaux, de sorte qu'un niveau d'accès, tel que ALTER, qui entraîne le contournement de toutes les vérifications entraîne le contournement des contrôles de sécurité pour tous les canaux.
  - Si vous exécutez CICS, vérifiez le paramètre RESSEC de la transaction.
  - Si RESLEVEL a été modifié alors qu'un utilisateur est connecté, il doit se déconnecter et se reconnecter avant que le nouveau paramètre RESLEVEL ne soit appliqué.
- Utilisez-vous des groupes de partage de files d'attente?

- Si vous utilisez la sécurité au niveau du groupe de partage de files d'attente et du gestionnaire de files d'attente, vérifiez que vous avez défini tous les profils corrects. Si le profil de gestionnaire de files d'attente n'est pas défini, un message indiquant que le profil est introuvable est envoyé au journal.
- Avez-vous utilisé une combinaison de paramètres de commutateur qui n'est pas valide pour que la vérification de la sécurité complète soit activée?
- Avez-vous besoin de définir des commutateurs de sécurité pour remplacer certains des paramètres de groupe de partage de files d'attente de votre gestionnaire de files d'attente?
- Un profil de niveau gestionnaire de files d'attente est-il prioritaire sur un profil de niveau groupe de partage de files d'attente?

z/OS Remarques relatives à la sécurité de l'initiateur de canal sous z/OS

Si vous utilisez la sécurité des ressources dans un environnement de mise en file d'attente répartie, l'espace adresse de l'initiateur de canal doit disposer d'un accès approprié aux différentes ressources IBM MQ . Vous pouvez utiliser la fonction ICSF (Integrated Cryptographic Support Facility) pour alimenter l'algorithme de protection par mot de passe.

Utilisation de la sécurité des ressources

Si vous utilisez la sécurité des ressources, tenez compte des points suivants si vous utilisez la mise en file d'attente répartie:

files d'attente système
L'espace adresse de l'initiateur de canal a besoin d'un accès RACF UPDATE aux files d'attente système répertoriées dans «Sécurité de la file d'attente système», à la page 209, ainsi qu'à toutes les files d'attente de destination utilisateur et à la file d'attente de rebut (voir «Sécurité de la file d'attente de rebut», à la page 208).

Files d'attente de transmission
L'espace adresse de l'initiateur de canal doit disposer d'un accès ALTER à toutes les files d'attente de transmission utilisateur.

sécurité du contexte
L'ID utilisateur du canal (et l'ID utilisateur MCA, le cas échéant) doit disposer de l'accès RACF CONTROL aux profils hlq.CONTEXT.queuename dans la classe MQADMIN. En fonction du profil RESLEVEL, l'ID utilisateur du canal peut également avoir besoin de l'accès CONTROL à ces profils.

Tous les canaux ont besoin de l'accès CONTROL au MQADMIN hlq.CONTEXT. profil de file d'attente de rebut. Tous les canaux (qu'ils soient initiateurs ou répondants) peuvent générer des rapports et, par conséquent, ils ont besoin de l'accès CONTROL au profil hlq.CONTEXT.reply-q .

Les canaux SENDER, CLUSSDR et SERVER ont besoin de l'accès CONTROL aux profils hlq.CONTEXT.xmit-queue-name car les messages peuvent être placés dans la file d'attente de transmission pour réveiller le canal et s'arrêter correctement.

Remarque : Si l'ID utilisateur de canal, ou un groupe RACF auquel l'ID utilisateur de canal est connecté, dispose d'un accès CONTROL ou ALTER à hlq.RESLEVEL, il n'y a aucune vérification des ressources pour l'initiateur de canal ou l'un de ses canaux.

Pour plus d'informations, voir «Profils pour la sécurité de contexte», à la page 221 «RESLEVEL et la connexion de l'initiateur de canal», à la page 242 et «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244 .

CSQINPX
Si vous utilisez le fichier d'entrée CSQINPX, l'initiateur de canal a également besoin d'un accès en lecture (READ) à CSQINPX et d'un accès en mise à jour (UPDATE) au fichier CSQOUTX et aux files d'attente dynamiques SYSTEM.CSQXCMD. *.

Sécurité des connexions
Les demandes de connexion de l'espace adresse de l'initiateur de canal utilisent le type de connexion CHIN, pour lequel la sécurité d'accès appropriée doit être définie. Voir «Profils de sécurité de connexion pour l'initiateur de canal», à la page 202.

Les ensembles de données

L'espace adresse de l'initiateur de canal doit disposer d'un accès approprié aux fichiers du gestionnaire de files d'attente. Voir «Autorisation de l'accès aux fichiers», à la page 261.

Commandes

Les commandes de mise en file d'attente répartie (par exemple, DEFINE CHANNEL, START CHINIT, START LISTENER et autres commandes de canal) doivent avoir la sécurité de commande appropriée définie, voir Tableau 49, à la page 224.

Si vous utilisez un groupe de partage de files d'attente, l'initiateur de canal peut émettre diverses commandes en interne, de sorte que l'ID utilisateur qu'il utilise doit être autorisé à émettre de telles commandes. Ces commandes sont START et STOP CHANNEL pour chaque canal utilisé avec CHLDISP (SHARED).

Si le PSMODE du gestionnaire de files d'attente n'est pas DISABLED, l'initiateur de canal doit disposer d'un accès en lecture (READ) à la commande DISPLAY PUBSUB.

Sécurité des canaux

Les canaux, en particulier les récepteurs et les connexions serveur, ont besoin d'une sécurité appropriée pour être configurés. Pour plus d'informations, voir «ID utilisateur pour le contrôle de sécurité sous z/OS», à la page 244 .

Vous pouvez également utiliser le protocole TLS (Transport Layer Security) pour assurer la sécurité sur les canaux. Pour plus d'informations sur l'utilisation de TLS avec IBM MQ, voir «Protocoles de sécurité TLS dans IBM MQ», à la page 24 .

Voir aussi «Contrôle d'accès pour les clients», à la page 99 pour plus d'informations sur la sécurité de la connexion serveur.

ID utilisateur

Les ID utilisateur décrits dans «ID utilisateur utilisés par l'initiateur de canal», à la page 248 et «ID utilisateur utilisés par l'agent de mise en file d'attente intra-groupe», à la page 252 ont besoin des accès suivants:

- Accès RACF UPDATE aux files d'attente de destination appropriées et à la file d'attente de rebut
- Accès RACF CONTROL au profil `hlq.CONTEXT.queuename` si la vérification du contexte est effectuée au niveau du récepteur
- Accès approprié à hlq.ALTERNATE.USER.userid qu'ils peuvent avoir besoin d'utiliser.
- Pour les clients, l'accès RACF approprié aux ressources à utiliser.

Sécurité APPC

Définissez la sécurité APPC appropriée si vous utilisez le protocole de transmission LU 6.2 . (Utilisez la classe APPCLU RACF par exemple.) Pour plus d'informations sur la configuration de la sécurité pour APPC, voir les manuels suivants:

- *z/OS V1R2.0 Planification MVS: Gestion APPC*
- *Multiplatform APPC Configuration Guide*, une publication IBM Redbooks

Les transmissions sortantes utilisent l'option APPC "SECURITY (SAME)" . Par conséquent, l'ID utilisateur de l'espace adresse de l'initiateur de canal et son profil par défaut (RACF GROUP) sont transmis sur le réseau au récepteur avec un indicateur indiquant que l'ID utilisateur a déjà été vérifié (ALREADYV).

Si le côté récepteur est également z/OS, l'ID utilisateur et le profil sont vérifiés par APPC et l'ID utilisateur est présenté au canal récepteur et utilisé comme ID utilisateur du canal.

Dans un environnement où le gestionnaire de files d'attente utilise APPC pour communiquer avec un autre gestionnaire de files d'attente sur le même système ou sur un autre système z/OS , vous devez vous assurer que:

- La définition VTAM de l'unité logique communicante spécifie SETACPT (ALREADYV)
- Il existe un profil APPCLU RACF pour la connexion entre les unités logiques qui spécifie CONVSEC (ALREADYV)

modification des paramètres de sécurité
Si le niveau d'accès RACF de l'ID utilisateur du canal ou de l'ID utilisateur MCA à une file d'attente de destination est modifié, cette modification ne prend effet que pour les nouveaux descripteurs d'objet (c'est-à-dire les nouveaux MQOPEN) de la file d'attente de destination. Les moments où les agents MCA ouvrent et ferment des files d'attente sont variables ; si un canal est déjà en cours d'exécution lorsqu'un tel changement d'accès est effectué, l'agent MCA peut continuer à placer des messages dans la file d'attente de destination en utilisant l'accès de sécurité existant des ID utilisateur plutôt que l'accès de sécurité mis à jour. L'arrêt et le redémarrage des canaux pour appliquer le niveau d'accès mis à jour permettent d'éviter ce scénario.

redémarrage automatique
Si vous utilisez z/OS Automatic Restart Manager (ARM) pour redémarrer l'initiateur de canal, l'ID utilisateur associé à l'espace adresse XCFAS doit être autorisé à émettre la commande IBM MQ START CHINIT.

Utilisation de la fonction ICSF (Integrated Cryptographic Service Facility)

L'initiateur de canal peut utiliser ICSF pour générer un nombre aléatoire lors de la distribution de l'algorithme de protection par mot de passe afin de brouiller les mots de passe transitant sur les canaux client si TLS n'est pas utilisé. Le processus de génération d'un nombre aléatoire est appelé *entropie*.

Si la fonction z/OS est installée mais que vous n'avez pas démarré ICSF, le message CSQX213E s'affiche et l'initiateur de canal utilise STCK pour l'entropie.

Le message CSQX213E vous avertit que l'algorithme de protection par mot de passe n'est pas aussi sécurisé qu'il pourrait l'être. Toutefois, vous pouvez poursuivre votre processus ; il n'y a pas d'autre impact sur l'exécution.

Si la fonction z/OS n'est pas installée, l'initiateur de canal utilise automatiquement STCK.

Remarques :

1. L'utilisation d'ICSF pour l'entropie génère plus de séquences aléatoires que l'utilisation de STCK.
2. Si vous démarrez ICSF, vous devez redémarrer l'initiateur de canal.
3. ICSF est requis pour certains CipherSpecs. Si vous tentez d'utiliser l'un de ces CipherSpecs et qu'ICSF n'est pas installé, vous recevez le message CSQX629E.

z/OS Sécurité dans les clusters de gestionnaires de files d'attente sous z/OS

Les considérations de sécurité pour les clusters sont les mêmes pour les gestionnaires de files d'attente et les canaux qui ne sont pas mis en cluster. L'initiateur de canal doit accéder à des files d'attente système supplémentaires et à des commandes supplémentaires nécessitant un ensemble de sécurité approprié.

Vous pouvez utiliser l'ID utilisateur MCA, les enregistrements d'authentification de canal, TLS et les exits de sécurité pour authentifier les canaux de cluster (comme avec les canaux conventionnels). Les enregistrements d'authentification de canal ou l'exit de sécurité relatif au canal récepteur de cluster doivent vérifier que le gestionnaire de files d'attente éloignées est autorisé à accéder aux files d'attente de cluster du gestionnaire de files d'attente de serveur. Vous pouvez commencer à utiliser la prise en charge du cluster IBM MQ sans changer la sécurité d'accès à la file d'attente existante. Vous devez cependant autoriser d'autres gestionnaires de files d'attente du cluster à écrire dans SYSTEM.CLUSTER.COMMAND.QUEUE si elles doivent rejoindre le cluster.

La prise en charge de cluster IBM MQ ne fournit pas de mécanisme permettant de limiter un membre d'un cluster au rôle client uniquement. Par conséquent, vous devez vous assurer que vous faites confiance aux gestionnaires de files d'attente que vous autorisez dans le cluster. Si un gestionnaire de files d'attente du cluster crée une file d'attente avec un nom particulier, il peut recevoir des messages pour cette file d'attente, que l'application ait ou non placé des messages dans cette file d'attente.

Pour restreindre l'appartenance à un cluster, effectuez la même action que celle que vous effectuez pour empêcher les gestionnaires de files d'attente de se connecter aux canaux récepteurs. Vous pouvez

restreindre l'appartenance d'un cluster en utilisant des enregistrements d'authentification de canal ou en écrivant un programme d'exit de sécurité sur le canal récepteur. Vous pouvez également écrire un programme d'exit pour empêcher les gestionnaires de files d'attente non autorisés d'écrire dans SYSTEM.CLUSTER.COMMAND.QUEUE.

Remarque : Il n'est pas conseillé d'autoriser les applications à ouvrir SYSTEM.CLUSTER.TRANSMIT.QUEUE directement. Il n'est pas non plus conseillé de permettre à une application d'ouvrir directement une autre file d'attente de transmission.

Si vous utilisez la sécurité des ressources, prenez en compte les points suivants en plus des considérations contenues dans «Remarques relatives à la sécurité de l'initiateur de canal sous z/OS», à la page 269:

files d'attente système
L'initiateur de canal a besoin de l'accès RACF ALTER aux files d'attente système suivantes:

- SYSTEME SYSTEM.CLUSTER.COMMAND
- SYSTEM.CLUSTER.TRANSMIT.QUEUE.

et accès UPDATE à SYSTEM.CLUSTER.REPOSITORY.QUEUE

Il a également besoin d'un accès en lecture (READ) à toutes les listes de noms utilisées pour la mise en cluster.

Commandes
Définissez la sécurité de commande appropriée (comme décrit dans Tableau 49, à la page 224) pour les commandes de prise en charge de cluster (REFRESH et RESET CLUSTER, SUSPEND et RESUME QMGR.

z/OS Remarques relatives à la sécurité pour l'utilisation de IBM MQ avec CICS

Toutes les versions de CICS prises en charge par IBM MQ 9.0.0 et les versions ultérieures utilisent la version fournie par CICS de l'adaptateur et de la passerelle.

Pour plus de détails sur les considérations de sécurité, voir:

- Sécurité de l'adaptateur CICS-IBM MQ.
- Sécurité du pont CICS-IBM MQ.

z/OS Remarques relatives à la sécurité pour l'utilisation de IBM MQ avec IMS

Utilisez cette rubrique pour planifier vos exigences de sécurité lorsque vous utilisez IBM MQ avec IMS.

Utilisation de la classe OPERCMDS

Si vous utilisez RACF pour protéger les ressources de la classe OPERCMDS, assurez-vous que l'ID utilisateur associé à l'espace adresse du gestionnaire de files d'attente IBM MQ est autorisé à émettre la commande MODIFY sur tout système IMS auquel il peut se connecter.

Remarques relatives à la sécurité pour le pont IMS

Vous devez prendre en compte quatre aspects lorsque vous déterminez vos exigences de sécurité pour le pont IMS :

- Quelle autorisation de sécurité est nécessaire pour connecter IBM MQ à IMS
- Niveau de contrôle de sécurité effectué sur les applications utilisant le pont pour accéder à IMS
- Les ressources IMS que ces applications sont autorisées à utiliser
- Droits à utiliser pour les messages insérés et reçus par le pont

Lorsque vous définissez vos exigences de sécurité pour le pont IMS , vous devez prendre en compte les éléments suivants:

- Les messages qui transitent par le pont peuvent provenir d'applications sur des plateformes qui n'offrent pas de fonctions de sécurité solides
- Les messages transmis via le pont peuvent provenir d'applications qui ne sont pas contrôlées par la même entreprise ou organisation

z/OS *Remarques relatives à la sécurité pour la connexion à IMS*

Accordez à l'ID utilisateur de l'espace adresse du gestionnaire de files d'attente IBM MQ l'accès au groupe OTMA.

Le pont IMS est un client OTMA. La connexion à IMS s'effectue sous l'ID utilisateur de l'espace adresse du gestionnaire de files d'attente IBM MQ . Il est normalement défini en tant que membre du groupe de tâches démarrées. Cet ID utilisateur doit être autorisé à accéder au groupe OTMA (sauf si le paramètre / SECURE OTMA est NONE).

Pour ce faire, définissez le profil suivant dans la classe FACILITY:

```
IMSXCF.xcfgname.mqxcfmname
```

Où `xcfgname` est le nom du groupe XCF et `mqxcfmname` est le nom du membre XCF de IBM MQ.

Vous devez accorder à votre ID utilisateur de gestionnaire de files d'attente IBM MQ un accès en lecture à ce profil.

Remarque :

1. Si vous modifiez les droits de la classe FACILITY, vous devez exécuter la commande RACF SETROPTS RACLIST (FACILITY) REFRESH pour activer les modifications.
2. Si le profil hlq.NO.SUBSYS.SECURITY existe dans la classe MQADMIN, aucun ID utilisateur n'est transmis à IMS et la connexion échoue sauf si le paramètre /SECURE OTMA est défini sur NONE.

z/OS *Contrôle d'accès aux applications pour le pont IMS*

Définissez un profil RACF dans la classe FACILITY pour chaque système IMS . Accordez un niveau d'accès approprié à l'ID utilisateur du gestionnaire de files d'attente IBM MQ .

Pour chaque système IMS auquel le pont IMS se connecte, vous pouvez définir le profil RACF suivant dans la classe FACILITY afin de déterminer le niveau de contrôle de sécurité effectué pour chaque message transmis au système IMS .

```
IMSXCF.xcfgname.imsxcfmname
```

Où `xcfgname` est le nom du groupe XCF et `imsxcfmname` est le nom du membre XCF pour IMS. (Vous devez définir un profil distinct pour chaque système IMS .)

Le niveau d'accès que vous autorisez pour l'ID utilisateur du gestionnaire de files d'attente IBM MQ dans ce profil est renvoyé à IBM MQ lorsque le pont IMS se connecte à IMSet indique le niveau de sécurité requis pour les transactions suivantes. Pour les transactions suivantes, IBM MQ demande les services appropriés à RACF et, lorsque l'ID utilisateur est autorisé, transmet le message à IMS.

OTMA ne prend pas en charge la commande IMS /SIGN ; toutefois, IBM MQ vous permet de définir la vérification d'accès pour chaque message afin d'activer l'implémentation du niveau de contrôle nécessaire.

Les informations de niveau d'accès suivantes peuvent être renvoyées:

NONE ou NO PROFILE FOUND

Ces valeurs indiquent que la sécurité maximale est requise, c'est-à-dire que l'authentification est requise pour chaque transaction. Une vérification est effectuée pour vérifier que l'ID utilisateur spécifié dans la zone *UserIdentifier* de la structure MQMD et le mot de passe ou PassTicket dans la zone *Authenticator* de la structure MQIIH sont connus de RACFet constituent une combinaison valide. UTOKEN est créé avec un mot de passe ou PassTicketet transmis à IMS ; UTOKEN n'est pas mis en cache.

Remarque : Si le profil hlq.NO.SUBSYS.SECURITY existe dans la classe MQADMIN, ce niveau de sécurité remplace celui défini dans le profil.

READ

Cette valeur indique que la même authentification doit être effectuée que pour NONE dans les cas suivants:

- La première fois qu'un ID utilisateur spécifique est rencontré
- Lorsque l'ID utilisateur a été rencontré auparavant mais que le jeton UTOKEN en cache n'a pas été créé avec un mot de passe ou PassTicket

IBM MQ demande un UTOKEN si nécessaire et le transmet à IMS.

Remarque : Si une demande de revérification de la sécurité a été traitée, toutes les informations mises en cache sont perdues et un UTOKEN est demandé la première fois que chaque ID utilisateur est détecté ultérieurement.

UPDATE

Il est vérifié que l'ID utilisateur figurant dans la zone *UserIdentifier* de la structure MQMD est connu de RACF.

Un UTOKEN est généré et transmis à IMS ; UTOKEN est mis en cache.

CONTROLE / ALTER

Ces valeurs indiquent qu'aucune UTOKENs de sécurité n'a besoin d'être fournie pour les ID utilisateur de ce système IMS . (Vous n'utiliseriez probablement cette option que pour les systèmes de développement et de test.)

Avertissement : Notez que l'ID utilisateur contenu dans la zone *UserIdentifier* de la structure MQMD est toujours transmis pour **CONTROL/ALTER**.

Remarque :

1. Cet accès est défini lorsque IBM MQ se connecte à IMSet dure pendant la durée de la connexion. Pour modifier le niveau de sécurité, l'accès au profil de sécurité doit être modifié, puis le pont doit être arrêté et redémarré (par exemple, en arrêtant et en redémarrant OTMA).
2. Si vous modifiez les droits de la classe FACILITY, vous devez exécuter la commande RACF SETROPTS RACLIST (FACILITY) REFRESH pour activer les modifications.
3. Vous pouvez utiliser un mot de passe ou un PassTicket, mais vous devez vous rappeler que le pont IMS ne chiffre pas les données. Pour plus d'informations sur l'utilisation de PassTickets, voir «Utilisation de RACF PassTickets dans l'en-tête IMS», à la page 276.
4. Certains de ces résultats peuvent être affectés par les paramètres de sécurité dans IMS, à l'aide de la commande /SECURE OTMA.
5. Les informations UTOKEN mises en cache sont conservées pendant la durée définie par les paramètres INTERVAL et TIMEOUT de la commande IBM MQ ALTER SECURITY.
6. L'option RACF WARNING n'a aucun effet sur le profil IMSXCF.xcfgname.imsxcfmname . Son utilisation n'affecte pas le niveau d'accès accordé et aucun message RACF WARNING n'est généré.

z/OS *Vérification de la sécurité sous IMS*

Les messages qui transitent par le pont contiennent des informations de sécurité. Les contrôles de sécurité effectués dépendent de la définition de la commande IMS /SECURE OTMA.

Chaque message IBM MQ qui passe par le pont contient les informations de sécurité suivantes:

- ID utilisateur contenu dans la zone *UserIdentifier* de la structure MQMD
- La portée de sécurité contenue dans la zone *SecurityScope* de la structure MQIIH (si la structure MQIIH est présente)
- Un UTOKEN (sauf si le sous-système IBM MQ dispose d'un accès CONTROL ou ALTER au profil `IMSXCF.xcfgname.imsxcfmname` approprié)

Les contrôles de sécurité effectués dépendent de la définition de la commande IMS /SECURE OTMA, comme suit:

/SECURE AUCUN OTMA
Aucun contrôle de sécurité n'est effectué pour la transaction.

/SECURE CONTROLE OTMA
La zone *UserIdentifier* de la structure MQMD est transmise à IMS pour vérification des droits d'accès aux transactions ou aux commandes.

Un élément ACEE (Accessor Environment Element) est généré dans la région de contrôle IMS .

/SECURE OTMA COMPLET
La zone *UserIdentifier* de la structure MQMD est transmise à IMS pour vérification des droits d'accès aux transactions ou aux commandes.

Un environnement ACEE est généré dans la région dépendante IMS ainsi que dans la région de contrôle IMS .

/PROFIL OTMA SECURISE
La zone *UserIdentifier* de la structure MQMD est transmise à IMS pour la vérification des droits de transaction ou de commande

La zone *SecurityScope* de la structure MQIIH est utilisée pour déterminer s'il convient de générer un ACEE dans la région dépendante IMS ainsi que dans la région de contrôle.

Remarque :

1. Si vous modifiez les droits de la classe TIMS ou CIMS ou des classes de groupe GIMS ou DIMSassociées, vous devez exécuter les commandes IMS suivantes pour activer les modifications:
   - /MODIFY PREPARE RACF
   - /MODIFIER VALIDATION
2. Si vous n'utilisez pas /SECURE OTMA PROFILE, toute valeur spécifiée dans la zone *SecurityScope* de la structure MQIIH est ignorée.

z/OS *Vérification de la sécurité effectuée par le pont IMS*

Des droits différents sont utilisés en fonction de l'action en cours.

Lorsque le pont insère ou reçoit un message, les droits suivants sont utilisés:

Obtention d'un message à partir de la file d'attente de pont
Aucun contrôle de sécurité n'est effectué.

Insertion d'une exception ou d'un message de rapport COA
Utilise les droits de l'ID utilisateur dans la zone *UserIdentifier* de la structure MQMD.

Insertion d'un message de réponse
Utilise les droits de l'ID utilisateur dans la zone *UserIdentifier* de la structure MQMD du message d'origine

Insertion d'un message dans la file d'attente de rebut
Aucun contrôle de sécurité n'est effectué.

Remarque :

1. Si vous modifiez les profils de classe IBM MQ , vous devez exécuter la commande IBM MQ REFRESH SECURITY (*) pour activer les modifications.

2. Si vous modifiez les droits d'un utilisateur, vous devez exécuter la commande MQSC RVERIFY SECURITY pour activer la modification.

z/OS *Utilisation de RACF PassTickets dans l'en-tête IMS*

Vous pouvez utiliser un PassTicket à la place d'un mot de passe dans l'en-tête IMS .

Si vous souhaitez utiliser un PassTicket à la place d'un mot de passe dans l'en-tête IMS (MQIIH), indiquez le nom de l'application par rapport à laquelle le PassTicket est validé dans l'attribut PASSTKTA de la définition STGCLASS de la file d'attente de pont IMS vers laquelle le message doit être acheminé.

Si la valeur PASSTKTA est laissée vide, vous devez faire en sorte qu'un PassTicket soit généré. Dans ce cas, le nom de l'application doit être au format MVSxxxx, où xxxx est le SMFID du système z/OS sur lequel s'exécute le gestionnaire de files d'attente cible.

Un PassTicket est généré à partir d'un ID utilisateur, du nom de l'application cible et d'une clé secrète. Il s'agit d'une valeur de 8 octets contenant des caractères alphabétiques et numériques en majuscules. Il ne peut être utilisé qu'une seule fois et est valide pour une période de 20 minutes. Si un PassTicket est généré par un système RACF local, RACF vérifie uniquement que le profil existe et non que l'utilisateur dispose de droits sur le profil. Si le PassTicket a été généré sur un système distant, RACF valide l'accès de l'ID utilisateur au profil. Pour plus d'informations sur les PassTickets, voir le manuel *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

PassTickets dans les en-têtes IMS sont donnés à RACF par IBM MQet non par IMS.

z/OS z/OS Migration d'un gestionnaire de files d'attente vers la sécurité à casse mixte

Procédez comme suit pour migrer un gestionnaire de files d'attente vers la sécurité à casse mixte. Vous vérifiez le niveau du produit de sécurité utilisé et activez les nouvelles classes de surveillance de sécurité externe IBM MQ. Exécutez la commande **REFRESH SECURITY** pour activer les profils à casse mixte.

Avant de commencer

1. Assurez-vous que toutes les classes de moniteur de sécurité externes d'IBM MQ sont activées.
2. Assurez-vous que votre gestionnaire de files d'attente est démarré.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour convertir un gestionnaire de files d'attente en sécurité à casse mixte.

Procédure

1. Copiez tous les profils et niveaux d'accès existants des classes majuscules dans la classe de surveillance de sécurité externe à casse mixte équivalente.
   a) `MQADMIN` dans `MXADMIN`.
   b) `MQPROC` dans `MXPROC`.
   c) `MQNLIST` dans `MXNLIST`.
   d) `MQQUEUE` dans `MXQUEUE`.
2. Remplacez la valeur de l'attribut SCYCASE par `MIXED`.

```
ALTER QMGR SCYCASE(MIXED)
```

3. Activez vos profils de sécurité existants.

```
REFRESH SECURITY(*) TYPE(CLASSES)
```

4. Vérifiez que vos profils de sécurité fonctionnent correctement.

Que faire ensuite

Passez en revue vos définitions d'objet et créez des profils à casse mixte, en utilisant **REFRESH SECURITY** selon les besoins pour activer les profils.

Configuration de la sécurité IBM MQ MQI client

Vous devez prendre en compte la sécurité IBM MQ MQI client pour que les applications client n'aient pas un accès illimité aux ressources sur le serveur.

Lors de l'exécution d'une application client, n'exécutez pas l'application à l'aide d'un ID utilisateur disposant de droits d'accès plus nombreux que nécessaire ; par exemple, un utilisateur du groupe `mqm` ou même l'utilisateur `mqm` lui-même.

En exécutant une application en tant qu'utilisateur disposant de trop de droits d'accès, vous risquez que l'application accède à des parties du gestionnaire de files d'attente et les modifie, soit par accident, soit par malveillance.

Il existe deux aspects de la sécurité entre une application client et son serveur de gestionnaire de files d'attente: l'authentification et le contrôle d'accès.

- L'authentification peut être utilisée pour s'assurer que l'application client, exécutée en tant qu'utilisateur spécifique, est bien celle qu'elle dit être. En utilisant l'authentification, vous pouvez empêcher un agresseur d'accéder à votre gestionnaire de files d'attente en empruntant l'une de vos applications.

  Depuis la IBM MQ 8.0, l'authentification est fournie par l'une des deux options suivantes:

  – La fonction d'authentification de connexion.

    Pour plus d'informations sur l'authentification de connexion, voir «Authentification de connexion», à la page 68.

  – Utilisation de l'authentification mutuelle dans TLS.

    Pour plus d'informations sur TLS, voir «Utilisation de SSL/TLS», à la page 282.

- Le contrôle d'accès peut être utilisé pour accorder ou supprimer des droits d'accès pour un utilisateur ou un groupe d'utilisateurs spécifique. En exécutant une application client avec un utilisateur créé spécifiquement (ou un utilisateur appartenant à un groupe spécifique), vous pouvez ensuite utiliser des contrôles d'accès pour vous assurer que l'application ne peut pas accéder à des parties de votre gestionnaire de files d'attente auxquelles l'application n'est pas censée accéder.

  Lors de la configuration du contrôle d'accès, vous devez tenir compte des règles d'authentification de canal et de la zone MCAUSER sur un canal. Ces deux fonctions permettent de modifier l'ID utilisateur utilisé pour vérifier les droits de contrôle d'accès.

  Pour plus d'informations sur le contrôle d'accès, voir «Autorisation de l'accès aux objets», à la page 360.

Si vous avez configuré une application client pour qu'elle se connecte à un canal spécifique avec un ID restreint, mais que le canal possède un ID administrateur défini dans sa zone MCAUSER, à condition que l'application client se connecte correctement, l'ID administrateur est utilisé pour les vérifications de contrôle d'accès. Par conséquent, l'application client dispose de droits d'accès complets à votre gestionnaire de files d'attente.

Pour plus d'informations sur l'attribut MCAUSER, voir «Mappage d'un ID utilisateur client à un ID utilisateur MCAUSER», à la page 400.

Les règles d'authentification de canal peuvent également être utilisées comme méthode de contrôle de l'accès à un gestionnaire de files d'attente, en définissant des règles et des critères spécifiques pour l'acceptation d'une connexion.

Pour plus d'informations sur les règles d'authentification de canal, voir: «Enregistrements d'authentification de canal», à la page 50.

Comment indiquer que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur MQI Client

Créez vos référentiels de clés à l'aide d'un logiciel compatible FIPS, puis indiquez que le canal doit utiliser des CipherSpecscertifiés FIPS.

Pour être conformes à la norme FIPS lors de l'exécution, les référentiels de clés doivent avoir été créés et gérés à l'aide de logiciels conformes à la norme FIPS tels que runmqakm avec l'option -fips.

Vous pouvez spécifier qu'un canal TLS doit utiliser uniquement des CipherSpecs certifiés FIPS de trois manières, répertoriées par ordre de priorité:

1. Définissez la zone FipsRequired dans la structure MQSCO sur MQSSL_FIPS_YES.
2. Définissez la variable d'environnement MQSSLFIPS sur YES.
3. Définissez l'attribut SSLFipsRequired sur YES dans le fichier de configuration du client.

Par défaut, les CipherSpecs certifiés FIPS ne sont pas requis.

Ces valeurs ont la même signification que les valeurs de paramètre équivalentes sur ALTER QMGR SSLFIPS (voir ALTER QMGR). Si le processus client ne possède actuellement aucune connexion TLS active et qu'une valeur FipsRequired est correctement spécifiée sur un MQCONNX SSL, toutes les connexions TLS suivantes associées à ce processus doivent utiliser uniquement les CipherSpecs associées à cette valeur. Cela s'applique jusqu'à l'arrêt de cette connexion et de toutes les autres connexions TLS, auquel cas une connexion MQCONNX ultérieure peut fournir une nouvelle valeur pour FipsRequired.

Si du matériel de cryptographie est présent, les modules de cryptographie utilisés par IBM MQ peuvent être configurés pour être ceux fournis par le produit matériel, et ceux-ci peuvent être certifiés FIPS à un niveau particulier. Les modules configurables et leur certification FIPS dépendent du produit matériel utilisé.

Dans la mesure du possible, si les CipherSpecs FIPS uniquement sont configurés, le client MQI rejette les connexions qui spécifient un CipherSpec non FIPS avec MQRC_SSL_INITIALIZATION_ERROR. IBM MQ ne garantit pas le rejet de toutes ces connexions et il est de votre responsabilité de déterminer si votre configuration IBM MQ est conforme à la norme FIPS.

Concepts associés

«FIPS (Federal Information Processing Standards) pour UNIX, Linux, and Windows», à la page 34
Lorsque la cryptographie est requise sur un canal SSL/TLS sur des systèmes Windows, UNIX and Linux , IBM MQ utilise un package de cryptographie appelé IBM Crypto for C (ICC). Sur les plateformes Windows, UNIX and Linux , le logiciel ICC a transmis le programme FIPS (Federal Information Processing Standards) Cryptomodule Validation Program du US National Institute of Standards and Technology, au niveau 140-2.

Strophe SSL du fichier de configuration client

Référence associée

FipsRequired (MQLONG)

MQSSLFIPS

AIX Exécution d'applications client TLS avec plusieurs installations de GSKit V8.0 sous AIX

Les applications client TLS sur AIX peuvent rencontrer des MQRC_CHANNEL_CONFIG_ERROR et des erreurs AMQ6175 lors de l'exécution sur des systèmes AIX avec plusieurs installations GSKit V8.0 .

Lors de l'exécution d'applications client sur un système AIX avec plusieurs installations GSKit V8.0 , les appels de connexion client peuvent renvoyer MQRC_CHANNEL_CONFIG_ERROR lors de l'utilisation de TLS. /var/mqm/errors consigne l'erreur d'enregistrement AMQ6175 et AMQ9220 pour l'application client défaillante, par exemple:

```
09/08/11 11:16:13 - Process(24412.1) User(user) Program(example)
```

```
Host(machine.example.ibm.com) Installation(Installation1)
VRMF(7.1.0.0)
AMQ6175: The system could not dynamically load the shared library
'/usr/mqm/gskit8/lib64/libgsk8ssl_64.so'. The system returned
error number '8' and error message 'Symbol resolution failed
for /usr/mqm/gskit8/lib64/libgsk8ssl_64.so because:
Symbol VALUE_EC_NamedCurve_secp256r1__9GSKASNOID (number 16) is not
exported from dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_NamedCurve_secp384r1__9GSKASNOID (number 17) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_NamedCurve_secp521r1__9GSKASNOID (number 18) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecPublicKey__9GSKASNOID (number 19) is not exported from dependent
module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecdsa_with_SHA1__9GSKASNOID (number 20) is not exported from
dependent module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.
Symbol VALUE_EC_ecdsa__9GSKASNOID (number 21) is not exported from dependent
module /db2data/db2inst1/sqllib/lib64/libgsk8cms_64.so.'.

EXPLANATION:
This message applies to AIX systems. The shared library
'/usr/mqm/gskit8/lib64/libgsk8ssl_64.so' failed
to load correctly due to a problem with the library.
ACTION:
Check the file access permissions and that the file has not been corrupted.
----- amqxufnx.c : 1284 ---------------------------------------------------------
09/08/11 11:16:13 - Process(24412.1) User(user) Program(example)
Host(machine.example.ibm.com) Installation(Installation1)
VRMF(7.1.0.0)
AMQ9220: The GSKit communications program could not be loaded.

EXPLANATION:
The attempt to load the GSKit library or procedure
'/usr/mqm/gskit8/lib64/libgsk8ssl_64.so' failed with error code
536895861.
ACTION:
Either the library must be installed on the system or the environment changed
to allow the program to locate it.
----- amqcgska.c : 836 ----------------------------------------------------------
```

Cette erreur est généralement due au fait que le paramètre de la variable d'environnement LIBPATH ou LD_LIBRARY_PATH a amené le client IBM MQ à charger un ensemble mixte de bibliothèques à partir de deux installations GSKit V8.0 différentes. L'exécution d'une application client IBM MQ dans un environnement Db2 peut provoquer cette erreur.

Pour éviter cette erreur, incluez les répertoires de la bibliothèque IBM MQ à l'avant du chemin de la bibliothèque afin que les bibliothèques IBM MQ soient prioritaires. Pour ce faire, utilisez la commande **setmqenv** avec le paramètre **-k** , par exemple:

```
. /usr/mqm/bin/setmqenv -s -k
```

Pour plus d'informations sur l'utilisation de la commande **setmqenv** , voir setmqenv (set IBM MQ environment)

IBM i Configuration des communications pour SSL ou TLS sur IBM i

Les communications sécurisées qui font appel aux protocoles de sécurité cryptographiques SSL ou TLS impliquent la configuration des canaux de communication et la gestion des certificats numériques à utiliser à des fins d'authentification.

Pour configurer votre installation SSL ou TLS, vous devez définir vos canaux pour utiliser les protocoles SSL ou TLS. Vous devez également créer et gérer vos certificats numériques. Sur certains systèmes d'exploitation, vous pouvez effectuer les tests avec des certificats autosignés. Toutefois, sous IBM i, vous devez utiliser des certificats personnels signés par une autorité de certification locale.

Pour plus d'informations sur la création et la gestion des certificats, voir «Utilisation de SSL/TLS sous IBM i», à la page 282.

Cette collection de rubriques présente certaines des tâches impliquées dans la configuration des communications SSL ou TLS et fournit des conseils étape par étape sur l'exécution de ces tâches.

Vous pouvez également tester l'authentification de client SSL ou TLS, qui sont des parties facultatives des protocoles SSL et TLS. Lors de l'établissement de liaison SSL ou TLS, le client SSL ou TLS obtient toujours un certificat numérique du serveur et le valide. Avec l'implémentation de IBM MQ, le serveur SSL ou TLS demande toujours un certificat au client.

Sous IBM i, le client SSL ou TLS envoie un certificat uniquement s'il en comporte un libellé au format IBM MQ correct:

- Pour un gestionnaire de files d'attente, `ibmwebspheremq` suivi du nom de votre gestionnaire de files d'attente est remplacé par des minuscules. Par exemple, pour `QM1`, `ibmwebspheremqqm1`.
- Pour un IBM MQ C Client for IBM i, `ibmwebspheremq` suivi de votre ID utilisateur de connexion remplacé par des minuscules, par exemple `ibmwebspheremqmyuserid`.

IBM MQ utilise le préfixe `ibmwebspheremq` sur un libellé pour éviter toute confusion avec les certificats d'autres produits. Veillez à spécifier l'intégralité du libellé de certificat en minuscules.

Le serveur SSL ou TLS valide toujours le certificat client si celui-ci est envoyé. Si le client SSL ou TLS n'envoie pas de certificat, l'authentification échoue uniquement si l'extrémité du canal agissant en tant que serveur SSL ou TLS est définie avec le paramètre SSLCAUTH défini sur REQUIRED ou une valeur de paramètre SSLPEER définie. Pour plus d'informations, voir Connexion de deux gestionnaires de files d'attente à l'aide de SSL ou TLS.

ULW Configuration des communications pour SSL ou TLS sur UNIX, Linux ou Windows

Les communications sécurisées qui font appel aux protocoles de sécurité cryptographiques SSL ou TLS impliquent la configuration des canaux de communication et la gestion des certificats numériques à utiliser à des fins d'authentification.

Pour configurer votre installation SSL ou TLS, vous devez définir vos canaux pour utiliser les protocoles SSL ou TLS. Vous devez également créer et gérer vos certificats numériques. Sur les systèmes UNIX, Linux et Windows , vous pouvez effectuer les tests avec des certificats autosignés.

Avertissement : Il n'est pas possible d'utiliser un mélange de certificats signés par Elliptic Curve et de certificats signés par RSA sur les gestionnaires de files d'attente que vous souhaitez joindre à l'aide de canaux activés par TLS.

Les gestionnaires de files d'attente utilisant des canaux TLS activés doivent tous utiliser des certificats signés par RSA ou tous utiliser des certificats signés par EC, et non les deux.

Pour plus d'informations, voir «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45.

Les certificats autosignés ne peuvent pas être révoqués, ce qui pourrait permettre à un agresseur de usurper une identité après qu'une clé privée a été compromise. Les autorités de certification peuvent révoquer un certificat compromis, pour en empêcher toute utilisation future. Les certificats signés par une autorité de certification sont donc plus sûrs à utiliser dans un environnement de production, bien que les certificats autosignés soient plus pratiques pour un système de test.

Pour plus d'informations sur la création et la gestion des certificats, voir «Utilisation de SSL/TLS sous UNIX, Linux, and Windows», à la page 294.

Cette collection de rubriques présente certaines des tâches liées à la configuration des communications SSL et fournit des conseils étape par étape sur l'exécution de ces tâches.

Vous pouvez également vouloir tester l'authentification des clients SSL ou TLS, qui constitue une partie facultative des protocoles. Lors de l'établissement de liaison SSL ou TLS, le client SSL ou TLS obtient toujours un certificat numérique du serveur et le valide. Avec l'implémentation de IBM MQ, le serveur SSL ou TLS demande toujours un certificat au client.

Sous UNIX, Linux, and Windows, le client SSL ou TLS envoie un certificat uniquement s'il en a un libellé au format IBM MQ correct:

- Pour un gestionnaire de files d'attente, le format est `ibmwebspheremq` suivi du nom de votre gestionnaire de files d'attente remplacé par des minuscules. Par exemple, pour `QM1`, `ibmwebspheremqqm1`
- Pour un client IBM MQ , `ibmwebspheremq` suivi de votre ID utilisateur de connexion est passé en minuscules, par exemple `ibmwebspheremqmyuserid`.

IBM MQ utilise le préfixe `ibmwebspheremq` sur un libellé pour éviter toute confusion avec les certificats d'autres produits. Veillez à spécifier l'intégralité du libellé de certificat en minuscules.

Le serveur SSL ou TLS valide toujours le certificat client si celui-ci est envoyé. Si le client n'envoie pas de certificat, l'authentification échoue uniquement si l'extrémité du canal agissant en tant que serveur SSL ou TLS est définie avec le paramètre SSLCAUTH défini sur REQUIRED ou une valeur de paramètre SSLPEER définie. Pour plus d'informations, voir Connexion de deux gestionnaires de files d'attente à l'aide de SSL ou TLS.

z/OS Configuration des communications pour SSL ou TLS sur z/OS

Les communications sécurisées qui font appel aux protocoles de sécurité cryptographiques SSL ou TLS impliquent la configuration des canaux de communication et la gestion des certificats numériques à utiliser à des fins d'authentification.

Pour configurer votre installation SSL ou TLS, vous devez définir vos canaux pour utiliser les protocoles SSL ou TLS. Vous devez également créer et gérer vos certificats numériques. Sous z/OS , vous pouvez effectuer les tests avec des certificats autosignés ou avec des certificats personnels signés par une autorité de certification locale.

Les certificats autosignés ne peuvent pas être révoqués, ce qui pourrait permettre à un agresseur de usurper une identité après qu'une clé privée a été compromise. Les autorités de certification peuvent révoquer un certificat compromis, pour en empêcher toute utilisation future. Les certificats signés par une autorité de certification sont donc plus sûrs à utiliser dans un environnement de production, bien que les certificats autosignés soient plus pratiques pour un système de test.

Pour plus d'informations sur la création et la gestion des certificats, voir «Utilisation de SSL/TLS sous z/OS», à la page 327.

Cette collection de rubriques présente certaines des tâches impliquées dans la configuration des communications SSL ou TLS et fournit des conseils étape par étape sur l'exécution de ces tâches.

Vous pouvez également vouloir tester l'authentification des clients SSL ou TLS, qui constitue une partie facultative des protocoles. Lors de l'établissement de liaison SSL ou TLS, le client SSL ou TLS obtient toujours un certificat numérique du serveur et le valide. Avec l'implémentation de IBM MQ, le serveur SSL ou TLS demande toujours un certificat au client.

Sous z/OS , le client SSL ou TLS envoie un certificat uniquement s'il possède l'un des certificats suivants:

- Pour un canal partagé uniquement, un certificat avec un libellé au format `ibmWebSphereMQ` suivi du nom de votre groupe de partage de files d'attente, par exemple `ibmWebSphereMQQSG1`
- Un certificat avec un libellé au format`ibmWebSphereMQ` suivi du nom de votre gestionnaire de files d'attente, par exemple `ibmWebSphereMQQM1`
- Un certificat par défaut (qui peut être le certificat `ibmWebSphereMQ`).

Si le canal est partagé, il tente d'abord de trouver un certificat pour le groupe de partage de files d'attente. S'il ne trouve pas de certificat pour un groupe de partage de files d'attente, il tente de trouver un certificat pour le gestionnaire de files d'attente.

Sous z/OS, IBM MQ utilise le préfixe `ibmWebSphereMQ` sur un libellé afin d'éviter toute confusion avec les certificats d'autres produits.

Le serveur SSL ou TLS valide toujours le certificat client si celui-ci est envoyé. Si le client SSL ou TLS n'envoie pas de certificat, l'authentification échoue uniquement si l'extrémité du canal agissant en tant que serveur SSL ou TLS est définie avec le paramètre SSLCAUTH défini sur REQUIRED ou une valeur de paramètre SSLPEER définie. Pour plus d'informations, voir Connexion de deux gestionnaires de files d'attente à l'aide de SSL ou TLS.

Utilisation de SSL/TLS

Ces rubriques fournissent des instructions pour l'exécution de tâches uniques liées à l'utilisation de TLS avec IBM MQ.

La plupart d'entre eux sont utilisés comme étapes dans les tâches de niveau supérieur décrites dans les sections suivantes:

- «Identification et authentification des utilisateurs», à la page 340
- «Autorisation de l'accès aux objets», à la page 360
- «Confidentialité des messages», à la page 433
- «Intégrité des données de messages», à la page 472
- «Maintien de la sécurité des clusters», à la page 473

IBM i Utilisation de SSL/TLS sous IBM i

Cette collection de rubriques fournit des instructions pour les tâches individuelles utilisant le protocole TLS (Transport Layer Security) dans IBM MQ for IBM i.

Pour IBM i , la prise en charge de TLS fait partie intégrante du système d'exploitation. Vérifiez que vous avez installé les prérequis répertoriés dans Configuration matérielle et logicielle requise sur IBM i.

Sous IBM i, vous gérez les clés et les certificats numériques à l'aide de l'outil Digital Certificate Manager (DCM).

Accès à DCM

Suivez ces instructions pour accéder à l'interface DCM.

Pourquoi et quand exécuter cette tâche

Effectuez les étapes suivantes dans un navigateur Web qui prend en charge les cadres.

Procédure

1. Accédez à `http://machine.domain:2001` ou à `https://machine.domain:2010`, où *machine* est le nom de votre ordinateur.
2. Entrez un profil utilisateur et un mot de passe valides lorsque vous y êtes invité.

   Vérifiez que votre profil utilisateur dispose des droits spéciaux *ALLOBJ et *SECADM pour vous permettre de créer de nouveaux magasins de certificats. Si vous ne disposez pas des droits spéciaux, vous pouvez uniquement gérer vos certificats personnels ou afficher les signatures d'objet pour les objets pour lesquels vous disposez de droits. Si vous êtes autorisé à utiliser une application de signature d'objet, vous pouvez également signer des objets à partir de DCM.
3. Sur la page Configurations Internet, cliquez sur **Gestionnaire de certificats Certificate Manager**.

   La page Certificate Manager numérique s'affiche.

Affectation d'un certificat à un gestionnaire de files d'attente sous IBM i

Utilisez DCM pour affecter un certificat à un gestionnaire de files d'attente.

Utilisez la gestion des certificats numériques IBM i pour affecter un certificat à un gestionnaire de files d'attente. Cela signifie que vous pouvez spécifier qu'un gestionnaire de files d'attente utilise le magasin de certificats de système et que le gestionnaire de files d'attente est enregistré pour être utilisé en tant qu'application avec le Certificate Managernumérique. Pour ce faire, remplacez la valeur de l'attribut **SSLKEYR** du gestionnaire de files d'attente par `*SYSTEM`.

Lorsque le paramètre **SSLKEYR** est remplacé par `*SYSTEM`, IBM MQ enregistre le gestionnaire de files d'attente en tant qu'application serveur avec un libellé d'application unique de QIBM_WEBSPHERE_MQ_QMGRNAME et un libellé avec une description de Qmgrname (WMQ). Notez que les attributs de canal **CERTLABL** ne sont pas utilisés si vous utilisez l'espace de stockage de certificats *SYSTEM. Le gestionnaire de files d'attente apparaît alors en tant qu'application serveur dans

le Certificate Managernumérique et vous pouvez affecter à cette application n'importe quel certificat serveur ou client dans le magasin système.

Etant donné que le gestionnaire de files d'attente est enregistré en tant qu'application, des fonctions avancées de DCM, telles que la définition de listes de confiance de l'autorité de certification, peuvent être exécutées.

Si le paramètre **SSLKEYR** est remplacé par une valeur autre que *SYSTEM, IBM MQ désenregistre le gestionnaire de files d'attente en tant qu'application avec Digital Certificate Manager. Si un gestionnaire de files d'attente est supprimé, il est également désenregistré de DCM. Un utilisateur disposant des droits *SECADM suffisants peut également ajouter ou supprimer manuellement des applications dans DCM.

Configuration d'un référentiel de clés sur IBM i

Un référentiel de clés doit être configuré aux deux extrémités de la connexion. Les magasins de certificats par défaut peuvent être utilisés ou vous pouvez créer les vôtres.

Une connexion TLS requiert un *référentiel de clés* à chaque extrémité de la connexion. Chaque gestionnaire de files d'attente et IBM MQ MQI client doivent avoir accès à un référentiel de clés. Si vous souhaitez accéder au référentiel de clés à l'aide d'un nom de fichier et d'un mot de passe (c'est-à-dire sans utiliser l'option *SYSTEM), vérifiez que le profil utilisateur QMQM dispose des droits suivants:

- Droit d'exécution sur le répertoire contenant le référentiel de clés
- Droits de lecture sur le fichier contenant le référentiel de clés

Pour plus d'informations, voir «Référentiel de clés SSL/TLS», à la page 25. Notez que les attributs **CERTLABL** de canal ne sont pas utilisés si vous utilisez l'espace de stockage de certificats *SYSTEM.

Sous IBM i, les certificats numériques sont stockés dans un magasin de certificats géré avec DCM. Ces certificats numériques comportent des libellés qui associent un certificat à un gestionnaire de files d'attente ou à un IBM MQ MQI client. TLS utilise les certificats à des fins d'authentification.

Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID de connexion de l'utilisateur IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.

Le nom du gestionnaire de files d'attente ou du magasin de certificats IBM MQ MQI client comprend un chemin et un nom de radical. Le chemin par défaut est `/QIBM/UserData/ICSS/Cert/Server/` et le nom de radical par défaut est `Default`. Sous IBM i, le magasin de certificats par défaut, `/QIBM/UserData/ICSS/Cert/Server/Default.kdb`, est également appelé `*SYSTEM`. Si vous le souhaitez, vous pouvez définir votre propre chemin et nom de radical.

Si vous définissez votre propre chemin ou nom de fichier, définissez les droits d'accès au fichier pour contrôler étroitement son accès.

«Modification de l'emplacement du référentiel de clés pour un gestionnaire de files d'attente sous IBM i», à la page 285 vous indique comment spécifier le nom du magasin de certificats. Vous pouvez spécifier le nom du magasin de certificats avant ou après la création du magasin de certificats.

Remarque : Les opérations que vous pouvez effectuer avec DCM peuvent être limitées par les droits de votre profil utilisateur. Par exemple, vous avez besoin des droits *ALLOBJ et *SECADM pour créer un certificat d'autorité de certification.

Création d'un magasin de certificats sous IBM i

Si vous ne souhaitez pas utiliser le magasin de certificats par défaut, suivez cette procédure pour créer le vôtre.

Pourquoi et quand exécuter cette tâche

Créez un nouveau magasin de certificats uniquement si vous ne souhaitez pas utiliser le magasin de certificats par défaut IBM i .

Pour indiquer que le magasin de certificats de système IBM i doit être utilisé, remplacez la valeur de l'attribut SSLKEYR du gestionnaire de files d'attente par *SYSTEM. Cette valeur indique que le

gestionnaire de files d'attente utilise le magasin de certificats du système et que le gestionnaire de files d'attente est enregistré pour être utilisé en tant qu'application avec Digital Certificate Manager (DCM).

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282
2. Dans le panneau de navigation, cliquez sur **Créer un nouveau magasin de certificats**.
   La page Créer un nouveau magasin de certificats s'affiche dans le cadre de la tâche.
3. Dans le cadre de la tâche, sélectionnez **Autre magasin de certificats de système** et cliquez sur **Continuer**.
   La page Créer un certificat dans le nouveau magasin de certificats s'affiche dans le cadre de la tâche.
4. Sélectionnez **Non-Ne pas créer de certificat dans le magasin de certificats** et cliquez sur **Continuer**.
   La page Nom et mot de passe du magasin de certificats s'affiche dans le cadre de la tâche.
5. Dans la zone **Chemin d'accès au magasin de certificats et nom de fichier** , entrez un chemin d'accès au système de fichiers intégré et un nom de fichier, par exemple `/QIBM/UserData/mqm/qmgrs/qm1/key.kdb`
6. Entrez un mot de passe dans la zone **Mot de passe** et entrez-le à nouveau dans la zone **Confirmer le mot de passe** . Cliquez sur **Continuer**.
   Notez le mot de passe (qui est sensible à la casse) car vous en avez besoin lorsque vous stockez la clé de référentiel.
7. Pour quitter DCM, fermez la fenêtre de votre navigateur.

Que faire ensuite

Une fois que vous avez créé le magasin de certificats à l'aide de DCM, veillez à stocker le mot de passe, comme décrit dans «Stockage du mot de passe du magasin de certificats sur les systèmes IBM i», à la page 284

Tâches associées

«Importation d'un certificat dans un référentiel de clés sous IBM i», à la page 290
Suivez cette procédure pour importer un certificat.

Stockage du mot de passe du magasin de certificats sur les systèmes IBM i
Stockez le mot de passe du magasin de certificats à l'aide de commandes CL.

Les instructions suivantes s'appliquent au stockage du mot de passe du magasin de certificats sur IBM i pour un gestionnaire de files d'attente. Sinon, pour un IBM MQ MQI client, si vous n'utilisez pas l'espace de stockage de certificats *SYSTEM (c'est-à-dire que l'environnement MQSSLKEYR est défini sur une valeur autre que *SYSTEM), suivez la procédure décrite dans la section «Stocker le mot de passe du magasin de certificats», à la page 293 de «IBM MQ Utilitaire de client SSL (amqrsslc) pour IBM i», à la page 292.

Si vous avez indiqué que l'espace de stockage de certificats *SYSTEM doit être utilisé (en remplaçant la valeur de l'attribut SSLKEYR du gestionnaire de files d'attente par *SYSTEM), vous ne devez pas suivre ces étapes.

Une fois que vous avez créé le magasin de certificats à l'aide de DCM, utilisez les commandes suivantes pour stocker le mot de passe:

```
STRMQM MQMNAME('queue_manager_name')
CHGMQM MQMNAME('queue_manager_name') SSLKEYRPWD('password')
```

Le mot de passe est sensible à la casse. Il doit être entré entre apostrophes exactement comme vous l'avez entré à l'étape 6 de la section «Création d'un magasin de certificats sous IBM i», à la page 283.

Remarque : Si vous n'utilisez pas le magasin de certificats de système par défaut et que vous ne stockez pas le mot de passe, les tentatives de démarrage des canaux TLS échouent car ils ne peuvent pas obtenir le mot de passe requis pour accéder au magasin de certificats.

Localisation du référentiel de clés d'un gestionnaire de files d'attente sous IBM i

Utilisez cette procédure pour obtenir l'emplacement du magasin de certificats de votre gestionnaire de files d'attente.

Procédure

1. Affichez les attributs de votre gestionnaire de files d'attente à l'aide de la commande suivante:

```
DSPMQM MQMNAME('queue manager name')
```

2. Examinez le résultat de la commande pour connaître le chemin et le nom de la racine du magasin de certificats.

   Par exemple: `/QIBM/UserData/ICSS/Cert/Server/Default`, où `/QIBM/UserData/ICSS/Cert/Server` est le chemin et `Default` le nom de la racine.

Modification de l'emplacement du référentiel de clés pour un gestionnaire de files d'attente sous IBM i

Modifiez l'emplacement du magasin de certificats de votre gestionnaire de files d'attente à l'aide de la commande CHGMQM ou ALTER QMGR.

Procédure

Utilisez la commande CHGMQM ou ALTER QMGR MQSC pour définir l'attribut de référentiel de clés de votre gestionnaire de files d'attente.

a) Utilisation de CHGMQM: `CHGMQM MQMNAME('qm1') SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')`

b) Utilisation de ALTER QMGR: `ALTER QMGR SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')`

Dans les deux cas, le magasin de certificats possède le nom de fichier complet: `/QIBM/UserData/ICSS/Cert/Server/MyKey.kdb`

Que faire ensuite

Lorsque vous modifiez l'emplacement du magasin de certificats d'un gestionnaire de files d'attente, les certificats ne sont pas transférés à partir de l'ancien emplacement. Si les certificats de l'autorité de certification préinstallés lorsque vous créez le magasin de certificats sont insuffisants, vous devez remplir le nouveau magasin de certificats avec des certificats, comme décrit dans «Importation d'un certificat dans un référentiel de clés sous IBM i», à la page 290. Vous devez également stocker le mot de passe pour le nouvel emplacement, comme décrit dans «Stockage du mot de passe du magasin de certificats sur les systèmes IBM i», à la page 284.

Création d'une autorité de certification et d'un certificat à des fins de test sur IBM i

Utilisez cette procédure pour créer un certificat d'autorité de certification local afin de signer des demandes de certificat et pour créer et installer le certificat d'autorité de certification.

Avant de commencer

Les instructions de cette rubrique supposent qu'une autorité de certification locale n'existe pas. S'il existe une autorité de certification locale, accédez à «Demande d'un certificat serveur sous IBM i», à la page 286.

Pourquoi et quand exécuter cette tâche

Les certificats de l'autorité de certification fournis lors de l'installation de TLS sont signés par l'autorité de certification émettrice. Sous IBM i, vous pouvez générer une autorité de certification locale qui peut signer des certificats de serveur pour tester les communications TLS sur votre système. Procédez comme suit dans un navigateur Web pour créer un certificat d'autorité de certification local:

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de navigation, cliquez sur **Créer une autorité de certification**.

   La page Créer une autorité de certification s'affiche dans le cadre de la tâche.
3. Entrez un mot de passe dans la zone **Mot de passe du magasin de certificats** et entrez-le à nouveau dans la zone **Confirmer le mot de passe** .
4. Entrez un nom dans la zone **Nom de l'autorité de certification** , par exemple `TLS Test Certificate Authority`.
5. Entrez les valeurs appropriées dans les zones **Nom usuel** et **Organisation** et sélectionnez un pays. Pour les autres zones facultatives, entrez les valeurs requises.
6. Entrez une période de validité pour l'autorité de certification locale dans la zone **Période de validité** .

   La valeur par défaut est 1095 jours.
7. Cliquez sur **Continuer**.

   L'autorité de certification est créée et DCM crée un magasin de certificats et un certificat d'autorité de certification pour votre autorité de certification locale.
8. Cliquez sur **Installer le certificat**.

   La boîte de dialogue du gestionnaire de téléchargement s'affiche.
9. Entrez le nom de chemin d'accès complet du fichier temporaire dans lequel vous souhaitez stocker le certificat de l'autorité de certification et cliquez sur **Sauvegarder**.
10. Une fois le téléchargement terminé, cliquez sur **Ouvrir**.

    La fenêtre Certificat s'affiche.
11. Cliquez sur **Installer le certificat**.

    L'assistant d'importation de certificat s'affiche.
12. Cliquez sur **Suivant**.
13. Sélectionnez **Sélectionner automatiquement le magasin de certificats en fonction du type de certificat** et cliquez sur **Suivant**.
14. Cliquez sur **Terminer**.

    Une fenêtre de confirmation s'affiche.
15. Cliquez sur **OK**.
16. Dans la fenêtre Certificat, cliquez sur **OK**.
17. Cliquez sur **Continuer**.

    La page Politique de l'autorité de certification s'affiche dans le cadre de la tâche.
18. Dans la zone **Autoriser la création de certificats d'utilisateur** , sélectionnez **Oui**.
19. Dans la zone **Période de validité** , entrez la période de validité des certificats émis par votre autorité de certification locale.

    La valeur par défaut est 365 jours.
20. Cliquez sur **Continuer**.

    La page Créer un certificat dans le nouveau magasin de certificats s'affiche dans le cadre de la tâche.
21. Vérifiez qu'aucune des applications n'est sélectionnée.
22. Cliquez sur **Continuer** pour terminer la configuration de l'autorité de certification locale.

Demande d'un certificat serveur sous IBM i

Les certificats numériques constituent une protection contre l'usurpation d'identité en certifiant qu'une clé publique appartient à une entité spécifiée. Un nouveau certificat serveur peut être demandé auprès d'une autorité de certification à l'aide du Certificate Manager numérique (DCM).

Pourquoi et quand exécuter cette tâche

Effectuez les étapes suivantes dans un navigateur Web:

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**.

   La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.
3. Sélectionnez le magasin de certificats que vous souhaitez utiliser et cliquez sur **Continuer**.
4. Facultatif : Si vous avez sélectionné ***SYSTEM** à l'étape 3, entrez le mot de passe du magasin système et cliquez sur **Continuer**.
5. Facultatif : Si vous avez sélectionné **Autre magasin de certificats système** à l'étape 3, dans la zone **Chemin d'accès au magasin de certificats et nom de fichier** , entrez le chemin et le nom de fichier IFS que vous avez définis lors de la création de votre magasin de certificats. Entrez également un mot de passe dans la zone **Certificate Store Password** . Cliquez ensuite sur **Continuer**
6. Dans le panneau de navigation, cliquez sur **Créer un certificat**.
7. Dans le cadre de la tâche, sélectionnez le bouton d'option **Serveur ou certificat client** et cliquez sur **Continuer**.

   La page Sélectionner une autorité de certification s'affiche dans le cadre de la tâche.
8. Si vous disposez d'une autorité de certification locale sur votre poste de travail, choisissez l'autorité de certification locale ou une autorité de certification commerciale pour signer le certificat. Sélectionnez le bouton d'option correspondant à l'autorité de certification de votre choix et cliquez sur **Continuer**.

   La page Créer un certificat s'affiche dans le cadre de la tâche.
9. Facultatif : Pour un gestionnaire de files d'attente, dans la zone **Libellé du certificat** , entrez le libellé du certificat.

   Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.

   Par exemple, pour le gestionnaire de files d'attente QM1, entrez `ibmwebspheremqqm1` pour utiliser la valeur par défaut.
10. Facultatif : Pour un IBM MQ MQI client, dans la zone **Libellé du certificat** , entrez `ibmwebspheremq` suivi de votre ID utilisateur de connexion en minuscules.
    Par exemple, saisissez : `ibmwebspheremqmyuserID`
11. Entrez les valeurs appropriées dans les zones **Nom usuel** et **Organisation** et sélectionnez un pays. Pour les autres zones facultatives, entrez les valeurs requises.

Résultats

Si vous avez sélectionné une autorité de certification commerciale pour signer votre certificat, DCM crée une demande de certificat au format PEM (Privacy-Enhanced Mail). Transmettez la demande à l'autorité de certification choisie.

Si vous avez sélectionné l'autorité de certification locale pour signer votre certificat, DCM vous informe que le certificat a été créé dans le magasin de certificats et qu'il peut être utilisé.

Demande d'un certificat serveur pour IBM Key Manager sous IBM i

Suivez cette procédure pour créer un certificat signé par votre autorité de certification locale ou pour demander un certificat serveur signé par une autorité de certification commerciale à importer dans l'utilitaire IBM Key Management (iKeyman).

Pourquoi et quand exécuter cette tâche

Un certificat d'utilisateur doit être utilisé lorsque le Certificate Manager (DCM) numérique sert de gestionnaire de certificats pour IBM MQ sur plusieurs plateformes. Pour les certificats personnels distribués à d'autres plateformes et pour les importer dans l'utilitaire iKeyman , effectuez les étapes suivantes dans un navigateur Web:

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de **navigation** , cliquez sur **Créer un certificat**.

   La page **Créer un certificat** s'affiche dans le cadre de la tâche.
3. Dans le panneau **Créer un certificat** , sélectionnez le bouton d'option **Certificat d'utilisateur** et cliquez sur **Continuer**.

   La page **Créer un certificat d'utilisateur** s'affiche.
4. Dans le panneau **Créer un certificat d'utilisateur** , renseignez les zones obligatoires sous Informations sur le certificat pour **Nom de l'organisation**, **Etat** ou province , **Pays** ou **région**. Vous pouvez éventuellement insérer des valeurs dans les zones **Unité organisationnelle** et **Localité** ou **ville** . Cliquez sur **Continuer**.

   Le **nom usuel** est automatiquement défini sur l'ID utilisateur avec lequel vous êtes connecté au système iSeries .
5. Dans le panneau **Create User Certificate** suivant, cliquez sur **Install certificate** , puis sur **Continue**.

   Un message s'affiche pour indiquer que `votre certificat personnel a été installé. Vous devez conserver une copie de sauvegarde de ce certificat.`
6. Cliquez sur **OK**.
7. Selon le navigateur Internet que vous avez utilisé pour accéder à DCM, procédez comme suit:
   a) Pour Microsoft Edge, choisissez: **Outils > Options Internet > Onglet Contenu > Bouton Certificats > Onglet Personnel >**. Sélectionnez le certificat et cliquez sur **Exporter**.
   b) Pour Mozilla Firefox, choisissez: **Outils > Options > Avancé > Onglet Chiffrement > Bouton Afficher les certificats > Onglet Vos certificats >**. Sélectionnez le certificat et cliquez sur **Sauvegarder**. Sélectionnez le chemin et le nom de fichier et cliquez sur **OK**.
8. Transférez le certificat exporté vers le système distant à l'aide de FTP au format binaire.
9. Ajoutez le certificat exporté de l'étape 7 à l'utilitaire iKeyman dans la base de données de clés.
   a) Si le certificat a été sauvegardé à l'aide de Microsoft Edge, utilisez les instructions décrites dans Importation à partir d'un fichier Microsoft `.pfx` .
   b) Si le certificat a été sauvegardé à l'aide de Mozilla Firefox, utilisez les instructions décrites dans Importation d'un certificat personnel dans un référentiel de clés.

   Lors de l'importation, vérifiez que le nom de libellé du certificat personnel et du certificat de signataire est remplacé par celui attendu par IBM MQ . Le libellé doit être soit la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.

Ajout de certificats serveur à un référentiel de clés sur IBM i

Suivez cette procédure pour ajouter un certificat demandé au référentiel de clés.

Pourquoi et quand exécuter cette tâche

Une fois que l'autorité de certification vous a envoyé un nouveau certificat serveur, vous l'ajoutez au magasin de certificats à partir duquel vous avez généré la demande. Si l'autorité de certification envoie le certificat dans le cadre d'un message électronique, copiez le certificat dans un fichier distinct.

Remarque :

- Vous n'avez pas besoin d'exécuter cette procédure si le certificat serveur est signé par votre autorité de certification locale.
- Avant d'importer un certificat serveur au format PKCS #12 dans DCM, vous devez d'abord importer le certificat de l'autorité de certification correspondant.

Utilisez la procédure suivante pour recevoir un certificat de serveur dans le magasin de certificats du gestionnaire de files d'attente:

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans la catégorie de tâche **Gérer les certificats** du panneau de navigation, cliquez sur **Importer un certificat**.

   La page Importer un certificat s'affiche dans le cadre de la tâche.
3. Sélectionnez le bouton d'option correspondant à votre type de certificat et cliquez sur **Continuer**.

   La page Importation d'un serveur ou d'un certificat client ou la page Importation d'un certificat d'autorité de certification s'affiche dans le cadre de la tâche.
4. Dans la zone **Importer un fichier** , entrez le nom de fichier du certificat à importer et cliquez sur **Continuer**.

   DCM détermine automatiquement le format du fichier.
5. Si le certificat est un certificat **serveur ou client** , entrez le mot de passe dans le cadre de la tâche et cliquez sur **Continuer**.

   DCM vous informe que le certificat a été importé.

Exportation d'un certificat à partir d'un référentiel de clés sous IBM i

L'exportation d'un certificat exporte à la fois la clé publique et la clé privée. Cette action doit être effectuée avec une extrême prudence, car la transmission d'une clé privée compromettrait complètement votre sécurité.

Avant de commencer

Lorsque vous partagez le certificat d'un utilisateur avec un autre utilisateur, vous échangez des clés publiques. Ce processus est décrit dans la tâche 5 de **. Certificats de partage** dans le Guide de démarrage rapide pour AMS sur UNIX. Lorsque vous exportez un certificat comme décrit ici, vous exportez à la fois la clé publique et la clé privée. Cette action doit être effectuée avec une extrême prudence, car la transmission d'une clé privée compromettrait complètement votre sécurité.

Pourquoi et quand exécuter cette tâche

Effectuez les étapes suivantes sur l'ordinateur à partir duquel vous souhaitez exporter le certificat:

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**.

   La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.
3. Sélectionnez le magasin de certificats que vous souhaitez utiliser et cliquez sur **Continuer**.
4. Facultatif : Si vous avez sélectionné ***SYSTEM** à l'étape 3, entrez le mot de passe du magasin système et cliquez sur **Continuer**.
5. Facultatif : Si vous avez sélectionné **Autre magasin de certificats système** à l'étape 3, dans la zone **Chemin et nom du magasin de certificats** , entrez le chemin et le nom de fichier IFS que vous avez définis lors de la création de votre magasin de certificats et entrez un mot de passe dans la zone **Mot de passe du magasin de certificats** . Cliquez ensuite sur **Continuer**
6. Dans la catégorie de tâche **Gérer les certificats** du panneau de navigation, cliquez sur **Exporter le certificat**.

   La page Exporter un certificat s'affiche dans le cadre de la tâche.
7. Sélectionnez le bouton d'option correspondant à votre type de certificat et cliquez sur **Continuer**.

   La page Exporter un certificat serveur ou client ou la page Exporter un certificat d'autorité de certification s'affiche dans le cadre de la tâche.
8. Sélectionnez le certificat à exporter.
9. Sélectionnez le bouton d'option pour indiquer si vous souhaitez exporter le certificat dans un fichier ou directement dans un autre magasin de certificats.

10. Si vous avez choisi d'exporter un certificat serveur ou client dans un fichier, fournissez les informations suivantes:
    - Chemin et nom de fichier de l'emplacement où vous souhaitez stocker le certificat exporté.
    - Pour un certificat personnel, mot de passe utilisé pour chiffrer le certificat exporté et l'édition cible. Pour les certificats de l'autorité de certification, vous n'avez pas besoin de spécifier le mot de passe.
11. Si vous avez choisi d'exporter un certificat directement dans un autre magasin de certificats, indiquez le magasin de certificats cible et son mot de passe.
12. Cliquez sur **Continuer**.

Importation d'un certificat dans un référentiel de clés sous IBM i

Suivez cette procédure pour importer un certificat.

Avant de commencer

Avant d'importer un certificat personnel au format PKCS #12 dans DCM, vous devez d'abord importer le certificat de l'autorité de certification correspondant.

Pourquoi et quand exécuter cette tâche

Effectuez ces étapes sur la machine sur laquelle vous souhaitez importer le certificat.

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**.

   La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.
3. Sélectionnez le magasin de certificats que vous souhaitez utiliser et cliquez sur **Continuer**.
4. Facultatif : Si vous avez sélectionné ***SYSTEM** à l'étape 3, entrez le mot de passe du magasin système et cliquez sur **Continuer**.
5. Facultatif : Si vous avez sélectionné **Autre magasin de certificats système** à l'étape 3, dans la zone **Chemin et nom du magasin de certificats** , entrez le chemin et le nom de fichier IFS que vous avez définis lors de la création de votre magasin de certificats et entrez un mot de passe dans la zone **Mot de passe du magasin de certificats** . Cliquez ensuite sur **Continuer**
6. Dans la catégorie de tâche **Gérer les certificats** du panneau de navigation, cliquez sur **Importer un certificat**.

   La page Importer un certificat s'affiche dans le cadre de la tâche.
7. Sélectionnez le bouton d'option correspondant à votre type de certificat et cliquez sur **Continuer**.

   La page d'importation du serveur ou du certificat client ou la page d'importation du certificat de l'autorité de certification s'affiche dans le cadre de la tâche.
8. Dans la zone **Importer un fichier** , entrez le nom de fichier du certificat à importer et cliquez sur **Continuer**.

   DCM détermine automatiquement le format du fichier.
9. Si le certificat est un certificat **serveur ou client** , entrez le mot de passe dans le cadre de la tâche et cliquez sur **Continuer**. DCM vous informe que le certificat a été importé.

Suppression de certificats dans IBM i

Utilisez cette procédure pour supprimer des certificats personnels.

Procédure

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**.

   La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.

3. Cochez la case **Autre magasin de certificats système** et cliquez sur **Continuer**.
   La page Magasin de certificats et mot de passe s'affiche.
4. Dans la zone **Chemin d'accès au magasin de certificats et nom de fichier** , entrez le chemin d'accès au système de fichiers intégré et le nom de fichier que vous avez définis lors de la création du magasin de certificats.
5. Entrez un mot de passe dans la zone **Certificate Store Password** . Cliquez sur **Continuer**.
   La page Magasin de certificats en cours s'affiche dans le cadre de la tâche.
6. Dans la catégorie de tâche **Gérer les certificats** du panneau de navigation, cliquez sur **Supprimer le certificat**.
   La page Confirmation de suppression de certificat s'affiche dans le cadre de la tâche.
7. Sélectionnez le certificat à supprimer. Cliquez sur **Supprimer**.
8. Cliquez sur **Oui** pour confirmer la suppression du certificat. Sinon, cliquez sur **Non**.
   DCM vous informe si le certificat a été supprimé.

*Utilisation de l'espace de stockage de certificats *SYSTEM pour l'authentification unidirectionnelle sous IBM i*

Suivez ces instructions pour configurer l'authentification unidirectionnelle.

Avant de commencer

- Créez un gestionnaire de files d'attente, des canaux et des files d'attente de transmission.
- Créez un certificat serveur ou client sur le gestionnaire de files d'attente du serveur.
- Transférez le certificat de l'autorité de certification au gestionnaire de files d'attente client et importez-le dans le référentiel de clés.
- Démarrez un programme d'écoute sur le serveur et les gestionnaires de files d'attente client.

Pourquoi et quand exécuter cette tâche

Pour utiliser l'authentification unidirectionnelle, en utilisant un ordinateur exécutant IBM i comme serveur TLS, définissez le paramètre SSL Key Repository (SSLKEYR) sur *SYSTEM. Ce paramètre enregistre le gestionnaire de files d'attente IBM MQ en tant qu'application. Vous pouvez ensuite affecter un certificat au gestionnaire de files d'attente pour activer l'authentification unidirectionnelle.

Vous pouvez également utiliser des magasins de clés privées pour implémenter l'authentification unidirectionnelle en créant un certificat factice pour le gestionnaire de files d'attente client dans le référentiel de clés.

Procédure

1. Effectuez les étapes suivantes sur les gestionnaires de files d'attente du serveur et du client:
   a) Modifiez le gestionnaire de files d'attente pour définir le paramètre SSLKEYR en exécutant la commande `CHGMQM MQMNAME(SSL) SSLKEYR(*SYSTEM)`.
   b) Stockez le mot de passe du référentiel de clés par défaut en exécutant la commande `CHGMQM MQMNAME(SSL) SSLKEYRPWD('xxxxxxx')`.
      Le mot de passe doit être entre apostrophes.
   c) Modifiez les canaux pour qu'ils aient le CipherSpec correct dans le paramètre SSLCIPHER.
   d) Actualisez la sécurité TLS en émettant la commande `RFRMQMAUT QMNAME(QMGRNAME) TYPE(*SSL)`.
2. Affectez le certificat au gestionnaire de files d'attente du serveur à l'aide de DCM, comme suit:
   a) Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
   b) Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**.
      La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.

c) Sélectionnez l'espace de stockage de certificats *SYSTEM et cliquez sur **Continuer**.

d) Dans le panneau de gauche, développez **Gérer les applications**.

e) Sélectionnez la définition **Afficher l'application** pour vérifier que le gestionnaire de files d'attente a été enregistré en tant qu'application.

SSL (WMQ) est répertorié dans le tableau.

f) Sélectionnez **Mettre à jour l'affectation de certificat**.

g) Sélectionnez **Serveur** et cliquez sur **Continuer**.

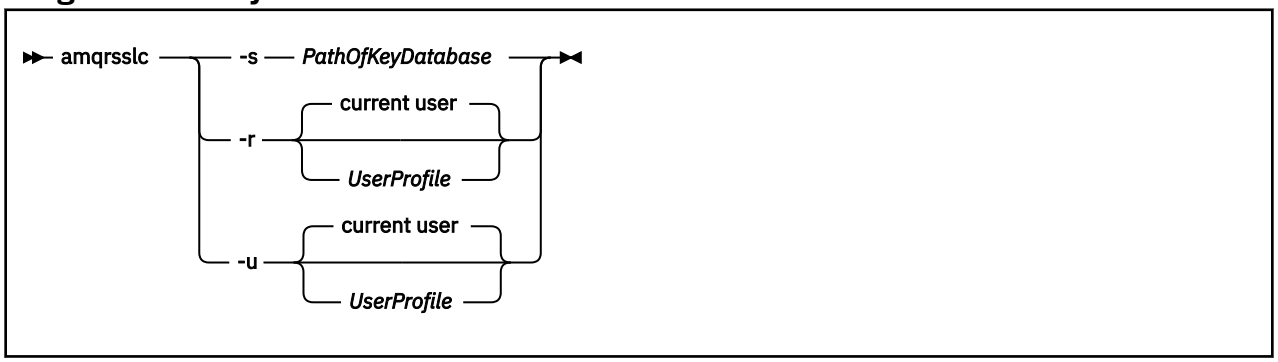
h) Sélectionnez QMGRNAME (WMQ) et cliquez sur **Mettre à jour l'affectation de certificat**.

i) Sélectionnez le certificat et cliquez sur **Affecter un nouveau certificat**. Une fenêtre s'ouvre pour indiquer que le certificat a été affecté à l'application.

IBM MQ Utilitaire de client SSL (amqrsslc) pour IBM i

L'utilitaire IBM MQ SSL Client (amqrsslc) for IBM i est utilisé par IBM MQ MQI client sur les systèmes IBM i pour enregistrer ou désenregistrer le profil utilisateur du client ou pour stocker le mot de passe du magasin de certificats. L'utilitaire ne peut être exécuté que par un utilisateur disposant des droits spéciaux *ALLOBJ ou par un membre de QMQMADM doté d'options permettant de créer ou de supprimer des enregistrements d'application dans le Certificate Manager numérique (DCM).

Diagramme de syntaxe



Enregistrer le profil utilisateur du client

Si IBM MQ MQI client utilise l'espace de stockage de certificats *SYSTEM, vous devez enregistrer le profil utilisateur du client (utilisateur connecté) pour l'utiliser en tant qu'application avec Digital Certificate Manager (DCM).

Si vous souhaitez enregistrer le profil utilisateur du client, exécutez le programme **amqrsslc** avec l'option `-r` avec *UserProfile*. Le profil utilisateur utilisé lors de l'appel de **amqrsslc** doit disposer du droit *USE. Si vous indiquez *UserProfile* avec l'option `-r` , l'option *UserProfile* est enregistrée en tant qu'application serveur avec un libellé d'application unique de QIBM_WEBSPHERE_MQ_*UserProfile* et un libellé avec une description de *UserProfile* (WMQ). Cette application serveur est ensuite affichée dans DCM et vous pouvez lui affecter n'importe quel certificat serveur ou client dans le magasin système.

Remarque : Si aucun profil utilisateur n'est indiqué avec l'option `-r` , le profil utilisateur de l'utilisateur exécutant l'outil **amqrsslc** est enregistré.

Le code suivant utilise **amqrsslc** pour enregistrer un profil utilisateur. Dans le premier exemple, le profil utilisateur spécifié est enregistré ; dans le second, il s'agit du profil de l'utilisateur connecté:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-r' UserProfile)
CALL PGM(QMQM/AMQRSSLC) PARM('-r')
```

Annulation de l'enregistrement du profil utilisateur du client

Pour annuler l'enregistrement du profil de client, exécutez le programme **amqrsslc** avec l'option -u avec *UserProfile*. Le profil utilisateur utilisé lors de l'appel de **amqrsslc** doit disposer du droit *USE. La fourniture de l'option *UserProfile* avec l'option -u désenregistre *UserProfile* avec le libellé QIBM_WEBSPHERE_MQ_*UserProfile* dans DCM.

Remarque : Si aucun profil utilisateur n'est spécifié avec l'option -u , le profil utilisateur de l'utilisateur exécutant l'outil **amqrsslc** est désenregistré.

Le code suivant utilise **amqrsslc** pour annuler l'enregistrement d'un profil utilisateur. Dans le premier exemple, le profil utilisateur spécifié est désenregistré ; dans le second, il s'agit du profil de l'utilisateur connecté:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-u' UserProfile)
CALL PGM(QMQM/AMQRSSLC) PARM('-u')
```

Stocker le mot de passe du magasin de certificats

Si IBM MQ MQI client n'utilise pas le magasin de certificats *SYSTEM et qu'il utilise un autre magasin de certificats (c'est-à-dire que MQSSLKEYR est défini sur une valeur autre que *SYSTEM), le mot de passe de la base de données de clés doit être stocké. Utilisez l'option -s pour stashing le mot de passe de la base de données de clés.

Dans le code suivant, le nom de fichier complet du magasin de certificats est /Path/Of/KeyDatabase/MyKey.kdb:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-s' '/Path/Of/KeyDatabase/MyKey')
```

L'exécution de ce code entraîne une demande de mot de passe pour cette base de données de clés. Ce mot de passe est stocké dans un fichier portant le même nom que la base de données de clés avec une extension .sth. Ce fichier est stocké dans le même chemin que la base de données de clés. L'exemple de code génère un fichier de dissimulation /Path/Of/KeyDatabase/MyKey.sth. QMQM est le propriétaire de l'utilisateur et QMQMADM le propriétaire du groupe pour ce fichier. QMQM et QMQMADM ont des droits de lecture, d'écriture et d'autres profils ont uniquement des droits de lecture.

Lorsque les modifications apportées aux certificats ou au magasin de certificats prennent effet sur IBM i

Lorsque vous modifiez les certificats dans un magasin de certificats ou l'emplacement du magasin de certificats, les modifications sont prises en compte en fonction du type de canal et de la manière dont le canal est en cours d'exécution.

Les modifications apportées aux certificats dans le magasin de certificats et à l'attribut de référentiel de clés prennent effet dans les situations suivantes:

- Lorsqu'un nouveau processus de canal unique sortant exécute pour la première fois un canal TLS.
- Lorsqu'un nouveau processus de canal unique TCP/IP entrant reçoit pour la première fois une demande de démarrage d'un canal TLS.
- Lorsque la commande MQSC REFRESH SECURITY TYPE (SSL) est émise pour actualiser l'environnement TLS IBM MQ .
- Pour les processus d'application client, lorsque la dernière connexion TLS du processus est fermée. La connexion TLS suivante récupère les changements de certificat.
- Pour les canaux qui s'exécutent en tant qu'unités d'exécution d'un processus de regroupement de processus (amqrmppa), lorsque le processus de regroupement de processus est démarré ou redémarré et exécute d'abord un canal TLS. Si le processus de regroupement de processus a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).

- Pour les canaux qui s'exécutent en tant qu'unités d'exécution de l'initiateur de canal, lorsque l'initiateur de canal est démarré ou redémarré et qu'il exécute d'abord un canal TLS. Si le processus initiateur de canal a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).
- Pour les canaux qui s'exécutent en tant qu'unités d'exécution d'un programme d'écoute TCP/IP, lorsque le programme d'écoute est démarré ou redémarré et qu'il reçoit d'abord une demande de démarrage d'un canal TLS. Si le programme d'écoute a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).

Configuration du matériel de cryptographie sous IBM i

Utilisez cette procédure pour configurer le coprocesseur de cryptographie sous IBM i

Avant de commencer

Vérifiez que votre profil utilisateur dispose des droits spéciaux *ALLOBJ et *SECADM pour vous permettre de configurer le matériel de coprocesseur.

Procédure

1. Accédez à `http://machine.domain:2001` ou à `https://machine.domain:2010`, où *machine* est le nom de votre ordinateur.

   Une boîte de dialogue s'affiche pour demander un nom d'utilisateur et un mot de passe.
2. Entrez un profil utilisateur et un mot de passe IBM i valides.
3. Accédez à Cryptographie et suivez les liens appropriés pour plus d'informations.

Que faire ensuite

Pour plus d'informations sur la configuration du 4767 Cryptographic Coprocessor, voir 4767 Cryptographic Coprocessor.

ULW Utilisation de SSL/TLS sous UNIX, Linux, and Windows

Sur les systèmes UNIX, Linux, and Windows , la prise en charge de TLS (Transport Layer Security) est installée avec IBM MQ.

Pour plus d'informations sur les règles de validation de certificat, voir Validation de certificat et conception de règles de confiance.

ULW *Utilisation de `runmqckm`, `runmqakm` et `strmqikm` pour gérer les certificats numériques*

Sur les systèmes UNIX, Linux, and Windows , gérez les clés et les certificats numériques avec **`strmqikm`** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide de **`runmqckm`** (iKeycmd) ou de **`runmqakm`** (GSKCapiCmd).

V 9.1.0

Avertissement : Les commandes **`runmqckm`** et **`strmqikm`** s'appuient sur l'environnement d'exécution Java (JRE) IBM MQ . Depuis IBM MQ 9.1, si l'environnement d'exécution Java n'est pas installé, vous recevez le message AMQ9183.

- Pour les systèmes **UNIX and Linux** :
  - Utilisez la commande **`strmqikm`** (iKeyman) pour démarrer l'interface graphique iKeyman .
  - Utilisez la commande **`runmqckm`** (iKeycmd) pour effectuer des tâches avec l'interface de ligne de commande iKeycmd .
  - Utilisez la commande **`runmqakm`** (GSKCapiCmd) pour effectuer des tâches avec l'interface de ligne de commande runmqakm. La syntaxe de commande de **`runmqakm`** est identique à celle de **`runmqckm`**.

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** à la place des commandes **runmqckm** ou **strmqikm** .

Voir Gestion des clés et des certificats pour une description complète des interfaces de ligne de commande pour les commandes **runmqckm** et **runmqakm** .

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que iKeycmd et iKeyman sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes iKeyman et iKeycmd sont 32 bits sur ces plateformes.

Pour plus d'informations, voir GSKit: PKCS#11 and IBM MQ JRE adressage mode .

Avant d'exécuter la commande **strmqikm** pour démarrer l'interface graphique d' iKeyman , vérifiez que vous travaillez sur une machine capable d'exécuter le système X-Window et que vous effectuez les opérations suivantes:

- Définissez la variable d'environnement DISPLAY, par exemple:

```
export DISPLAY=mypc:0
```

- Vérifiez que votre variable d'environnement PATH contient **/usr/bin** et **/bin**. Cette opération est également requise pour les commandes **runmqckm** et **runmqakm** . Exemple :

```
export PATH=$PATH:/usr/bin:/bin
```

- Pour les systèmes **Windows** :
  - Utilisez la commande **strmqikm** pour démarrer l'interface graphique iKeyman .
  - La commande **runmqckm** permet d'effectuer des tâches à l'aide de l'interface de ligne de commande iKeycmd .

    Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** à la place des commandes **runmqckm** ou **strmqikm** .
  - Utilisez la commande **runmqakm -keydb** avec l'option *stashpw* ou *stash* .

    Lorsque vous utilisez la commande **runmqakm -keydb** de cette manière, par exemple:

```
runmqakm -keydb -create -db key.kdb -pw secretpwd -stash
```

    le droit de lecture du fichier `.sth` résultant n'est pas activé pour le groupe `mqm` .

    Seul le créateur peut lire le fichier. Après avoir créé un fichier de dissimulation à l'aide de la commande **runmqakm** , vérifiez les droits d'accès au fichier et accordez des droits d'accès au compte de service exécutant le gestionnaire de files d'attente ou à un groupe tel que `mqm`local.

Pour demander le traçage TLS sur les systèmes UNIX, Linux ou Windows , voir strmqtrc.

Référence associée

Commandes runmqckm et runmqakm
Cette section décrit les commandes runmqckm et runmqakm en fonction de l'objet de la commande.

ULW *Configuration d'un référentiel de clés sur UNIX, Linux, and Windows*

Vous pouvez configurer un référentiel de clés à l'aide de **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd).

Pourquoi et quand exécuter cette tâche

Une connexion TLS requiert un *référentiel de clés* à chaque extrémité de la connexion. Chaque gestionnaire de files d'attente IBM MQ et IBM MQ MQI client doivent avoir accès à un référentiel de clés. Pour plus d'informations, voir «Référentiel de clés SSL/TLS», à la page 25.

Sur les systèmes UNIX, Linux, and Windows , les certificats numériques sont stockés dans un fichier de base de données de clés géré à l'aide de l'interface utilisateur **strmqikm** ou des commandes **runmqckm** ou **runmqakm** . Ces certificats numériques comportent des libellés. Un libellé spécifique associe un certificat personnel à un gestionnaire de files d'attente ou à IBM MQ MQI client. TLS utilise ce certificat à des fins d'authentification. Sur les systèmes UNIX, Linux, and Windows , IBM MQ utilise la valeur de l'attribut **CERTLABL** , si elle est définie, ou la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID de connexion utilisateur IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.

Le nom du fichier de la base de données de clés comprend un chemin et un nom de radical:

- Sur les systèmes UNIX and Linux , le chemin par défaut d'un gestionnaire de files d'attente (défini lors de la création du gestionnaire de files d'attente) est `/var/mqm/qmgrs/`*`queue_manager_name`*`/ssl`.

  Sur les systèmes Windows , le chemin par défaut est *`MQ_INSTALLATION_PATH`*`\Qmgrs\`*`queue_manager_name`*`\ssl`, où *`MQ_INSTALLATION_PATH`* est le répertoire dans lequel IBM MQ est installé. Par exemple, `C:\Program Files\IBM\MQ\Qmgrs\QM1\ssl`.

  Le nom de radical par défaut est `key`. Vous pouvez éventuellement choisir votre propre chemin et nom de radical, mais l'extension doit être `.kdb`.

  Si vous choisissez votre propre chemin ou nom de fichier, définissez les droits d'accès au fichier pour contrôler étroitement l'accès à ce dernier.
- Pour un client IBM MQ , il n'existe pas de chemin ou de nom de radical par défaut. Contrôler étroitement l'accès à ce fichier. L'extension doit être `.kdb`.

Ne créez pas de référentiels de clés sur un système de fichiers qui ne prend pas en charge les verrous de niveau fichier, par exemple NFS version 2 sur les systèmes Linux .

Pour plus d'informations sur la vérification et la spécification du nom de fichier de la base de données de clés, voir «Modification de l'emplacement du référentiel de clés pour un gestionnaire de files d'attente sous UNIX, Linux, and Windows», à la page 301 . Vous pouvez spécifier le nom du fichier de la base de données de clés avant ou après la création du fichier de la base de données de clés.

L'ID utilisateur à partir duquel vous exécutez les commandes **strmqikm** ou **runmqckm** doit disposer de droits d'accès en écriture pour le répertoire dans lequel le fichier de base de données de clés est créé ou mis à jour. Pour un gestionnaire de files d'attente utilisant le répertoire `ssl` par défaut, l'ID utilisateur à partir duquel vous exécutez **strmqikm** ou **runmqckm** doit être membre du groupe mqm. Pour un IBM MQ MQI client, si vous exécutez **strmqikm** ou **runmqckm** à partir d'un ID utilisateur différent de celui sous lequel le client s'exécute, vous devez modifier les droits d'accès aux fichiers pour permettre à IBM MQ MQI client d'accéder au fichier de la base de données de clés lors de l'exécution. Pour plus d'informations, voir «Accès et sécurisation de vos fichiers de la base de données clé sous Windows», à la page 298 ou «Accès et sécurisation de vos fichiers de la base de données clé sous les systèmes UNIX and Linux», à la page 298.

Dans **strmqikm** ou **runmqckm** for IBM WebSphere MQ 7.0, les nouvelles bases de données de clés sont automatiquement remplies avec un ensemble de certificats d'autorité de certification prédéfinis. Dans **strmqikm** ou **runmqckm** for IBM MQ 8.0, les bases de données de clés ne sont pas automatiquement remplies, ce qui rend la configuration initiale plus sécurisée car vous incluez uniquement les certificats de l'autorité de certification de votre choix dans votre fichier de base de données de clés.

Remarque : En raison de ce changement de comportement pour GSKit 8.0 qui a pour conséquence que les certificats de l'autorité de certification ne sont plus ajoutés automatiquement au référentiel, vous devez ajouter manuellement vos certificats de l'autorité de certification préférés. Ce changement de comportement vous offre un contrôle plus granulaire sur les certificats de l'autorité de certification

utilisés. Voir «Ajout de certificats d'autorité de certification par défaut dans un référentiel de clés vide sous UNIX, Linux, and Windows avec GSKit 8.0», à la page 299.

Vous créez la base de données de clés à l'aide de la ligne de commande ou de l'interface utilisateur **strmqikm** (iKeyman).

Remarque : Si vous devez gérer les certificats TLS d'une manière compatible avec FIPS, utilisez la commande **runmqakm** . L'interface utilisateur **strmqikm** ne fournit pas d'option compatible FIPS.

Procédure

Créez une base de données de clés à l'aide de la ligne de commande.

1. Exécutez l'une des commandes suivantes:
   - A l'aide de **runmqckm**:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - A l'aide de **runmqakm**:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
     -stash -fips -strong
     ```

   où :

   -db *nom_fichier*
   Indique le nom de fichier qualifié complet d'une base de données de clés CMS et doit avoir l'extension de fichier .kdb.

   -pw *mot_de_passe*
   Indique le mot de passe de la base de données de clés CMS.

   -type *cms*
   Indique le type de base de données. (Pour IBM MQ, il doit s'agir de cms.)

   -stash
   Sauvegarde le mot de passe de la base de données de clés dans un fichier.

   -fips
   indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

   -forte
   Vérifie que le mot de passe entré répond aux exigences minimales de puissance de mot de passe. Les exigences minimales pour un mot de passe sont les suivantes:

   - Le mot de passe doit avoir une longueur minimale de 14 caractères.
   - Le mot de passe doit contenir au moins un caractère minuscule, un caractère majuscule et un chiffre ou un caractère spécial. Les caractères spéciaux incluent l'astérisque (*), le signe dollar ($), le signe nombre (#) et le signe pourcentage (%). Un espace est classé comme un caractère spécial.
   - Chaque caractère peut apparaître au maximum trois fois dans un mot de passe.
   - Un maximum de deux caractères consécutifs dans le mot de passe peut être identique.
   - Tous les caractères se trouvent dans le jeu de caractères ASCII imprimables standard, compris entre 0x20 et 0x7E.

Vous pouvez également créer une base de données de clés à l'aide de l'interface utilisateur **strmqikm** (iKeyman).

2. Sur les systèmes UNIX and Linux , connectez-vous en tant que superutilisateur. Sur les systèmes Windows , connectez-vous en tant qu'administrateur ou en tant que membre du groupe MQM.
3. Démarrez l'interface utilisateur en exécutant la commande **strmqikm** .

4. Dans le menu **Fichier de base de données de clés** , cliquez sur **Nouveau**.

   La fenêtre Nouveau s'ouvre.
5. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
6. Dans la zone **Nom de fichier** , entrez un nom de fichier.

   Cette zone contient déjà le texte `key.kdb`. Si votre nom de radical est `key`, laissez cette zone inchangée. Si vous avez spécifié un autre nom de radical, remplacez `key` par votre nom de radical. Toutefois, vous ne devez pas modifier l'extension `.kdb` .
7. Dans la zone **Emplacement** , entrez le chemin d'accès.

   Exemple :

   - Pour un gestionnaire de files d'attente: `/var/mqm/qmgrs/QM1/ssl` (sur les systèmes UNIX and Linux) ou `C:\ProgramData\IBM\MQ\qmgrs\QM1\ssl` (sur les systèmes Windows).

     Le chemin doit correspondre à la valeur de l'attribut **SSLKeyRepository** du gestionnaire de files d'attente.
   - Pour un client IBM MQ : `/var/mqm/ssl` (sur les systèmes UNIX and Linux) ou `C:\mqm\ssl` (sur les systèmes Windows).
8. Cliquez sur **OK**.

   La fenêtre Password Prompt s'ouvre.
9. Entrez un mot de passe dans la zone **Mot de passe** , puis entrez-le à nouveau dans la zone **Confirmer le mot de passe** .
10. Cochez la case **Stocker le mot de passe dans un fichier** .

    Remarque : Si vous ne stockez pas le mot de passe, les tentatives de démarrage des canaux TLS échouent car ils ne peuvent pas obtenir le mot de passe requis pour accéder au fichier de la base de données de clés.
11. Cliquez sur **OK**.

    La fenêtre Certificats personnels s'ouvre.
12. Définissez les droits d'accès comme décrit dans «Accès et sécurisation de vos fichiers de la base de données clé sous Windows», à la page 298 ou «Accès et sécurisation de vos fichiers de la base de données clé sous les systèmes UNIX and Linux», à la page 298.

Windows *Accès et sécurisation de vos fichiers de la base de données clé sous Windows*

Il se peut que les fichiers de la base de données de clés ne disposent pas des droits d'accès appropriés. Vous devez définir l'accès approprié à ces fichiers.

Définissez le contrôle d'accès aux fichiers *key*`.kdb`, *key*`.sth`, *key*`.crl`et *key*`.rdb`, où *key* est le nom de radical de votre base de données de clés, pour accorder des droits à un ensemble restreint d'utilisateurs.

Envisagez d'accorder l'accès comme suit:

droits complets
: BUILTIN\Administrators, NT AUTHORITY\SYSTEM et l'utilisateur qui a créé les fichiers base de données.

droit de lecture
: Pour un gestionnaire de files d'attente, le groupe mqm local uniquement. Cela suppose que l'agent MCA s'exécute sous un ID utilisateur dans le groupe mqm.

  Pour un client, ID utilisateur sous lequel le processus client est exécuté.

Linux UNIX *Accès et sécurisation de vos fichiers de la base de données clé sous les systèmes UNIX and Linux*

Il se peut que les fichiers de la base de données de clés ne disposent pas des droits d'accès appropriés. Vous devez définir l'accès approprié à ces fichiers.

Pour un gestionnaire de files d'attente, définissez les droits d'accès aux fichiers de la base de données de clés de sorte que les processus de gestionnaire de files d'attente et de canal puissent les lire si nécessaire, mais que les autres utilisateurs ne puissent pas les lire ou les modifier. Normalement, l'utilisateur mqm a besoin de droits d'accès en lecture. Si vous avez créé le fichier de la base de données de clés en vous connectant en tant qu'utilisateur mqm, les droits sont probablement suffisants ; si vous n'étiez pas l'utilisateur mqm, mais un autre utilisateur du groupe mqm, vous devrez probablement accorder des droits de lecture à d'autres utilisateurs du groupe mqm.

De même pour un client, définissez des droits sur les fichiers de la base de données de clés afin que les processus de l'application client puissent les lire lorsque cela est nécessaire, mais les autres utilisateurs ne peuvent pas les lire ou les modifier. Normalement, l'utilisateur sous lequel le processus client s'exécute a besoin de droits de lecture. Si vous avez créé le fichier de la base de données de clés en vous connectant en tant que cet utilisateur, les droits sont probablement suffisants ; si vous n'étiez pas l'utilisateur du processus client, mais un autre utilisateur de ce groupe, vous devrez probablement accorder des droits de lecture à d'autres utilisateurs du groupe.

Définissez les droits sur les fichiers *key*.kdb, *key*.sth, *key*.crlet *key*.rdb, où *key* est le nom de radical de votre base de données de clés, pour lire et écrire pour le propriétaire du fichier, et pour lire pour le groupe d'utilisateurs mqm ou client (-rw-r -----).

ULW *Ajout de certificats d'autorité de certification par défaut dans un référentiel de clés vide sous UNIX, Linux, and Windows avec GSKit 8.0*

Suivez cette procédure pour ajouter un ou plusieurs des certificats de l'autorité de certification par défaut à un référentiel de clés vide avec GSKit version 8.

Dans GSKit 7.0, le comportement lors de la création d'un nouveau référentiel de clés consistait à ajouter automatiquement un ensemble de certificats d'autorité de certification par défaut pour les autorités de certification utilisées couramment. Pour GSKit version 8, ce comportement a été modifié de sorte que les certificats de l'autorité de certification ne sont plus automatiquement ajoutés au référentiel. L'utilisateur est désormais tenu d'ajouter manuellement des certificats de l'autorité de certification dans le référentiel de clés.

Utilisation `strmqikm`

Suivez les étapes suivantes sur la machine sur laquelle vous souhaitez ajouter le certificat de l'autorité de certification :

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** (sous UNIX, Linux, and Windows).
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de clés dans lequel vous souhaitez ajouter le certificat, par exemple key.kdb.
6. Cliquez sur **Ouvrir**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de clés s'affiche dans la zone **Nom du fichier**.
8. Dans la zone **Key database content**, sélectionnez **Signer Certificates**.
9. Cliquez sur **Remplir**. La fenêtre Ajouter un certificat de l'autorité de certification s'ouvre.
10. Les certificats de l'autorité de certification pouvant être ajoutés au référentiel sont affichés dans une structure d'arborescence hiérarchique. Sélectionnez l'entrée de niveau supérieur pour l'organisation dont vous souhaitez faire confiance aux certificats de l'autorité de certification pour afficher la liste complète des certificats de l'autorité de certification valides.
11. Sélectionnez dans la liste les certificats de l'autorité de certification que vous souhaitez approuver et cliquez sur **OK**. Les certificats sont ajoutés au référentiel de clés.

Utilisation de la ligne de commande

Utilisez les commandes suivantes pour répertorier, puis ajoutez des certificats d'autorité de certification à l'aide de **runmqckm**:

- Exécutez la commande suivante pour répertorier les certificats de l'autorité de certification par défaut avec les organisations qui les émettent:

```
runmqckm -cert -listsigners
```

- Exécutez la commande suivante pour ajouter tous les certificats de l'autorité de certification pour l'organisation spécifiée dans la zone *label* :

```
runmqckm -cert -populate -db filename -pw password -label label
```

où :

-db *filename*	est le nom de chemin qualifié complet de la base de données de clés.
-pw *password*	est le mot de passe de la base de données de clés.
-label *label*	correspond au libellé du certificat.

Remarque : L'ajout d'un certificat d'autorité de certification à un référentiel de clés permet à IBM MQ de faire confiance à tous les certificats personnels signés par ce certificat d'autorité de certification. Réfléchissez soigneusement aux autorités de certification que vous souhaitez accréditer et ajoutez uniquement l'ensemble de certificats de l'autorité de certification requis pour authentifier vos clients et vos gestionnaires. Il n'est pas recommandé d'ajouter l'ensemble complet des certificats de l'autorité de certification par défaut sauf s'il s'agit d'une exigence définitive pour votre politique de sécurité.

ULW *Localisation du référentiel de clés d'un gestionnaire de files d'attente sous UNIX, Linux, and Windows*

Utilisez cette procédure pour obtenir l'emplacement du fichier de base de données de clés de votre gestionnaire de files d'attente

Procédure

1. Affichez les attributs de votre gestionnaire de files d'attente à l'aide de l'une des commandes MQSC suivantes:

```
DISPLAY QMGR ALL
DISPLAY QMGR SSLKEYR
```

   Vous pouvez également afficher les attributs de votre gestionnaire de files d'attente à l'aide des commandes IBM MQ Explorer ou PCF.
2. Recherchez dans le résultat de la commande le chemin et le nom de la racine du fichier de la base de données de clés.
   Par exemple :
   a. sous UNIX and Linux: `/var/mqm/qmgrs/QM1/ssl/key`, où `/var/mqm/qmgrs/QM1/ssl` est le chemin d'accès et `key` est le nom de la racine
   b. sous Windows: *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl\key`, où *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl` est le chemin et `key` est le nom de la racine. *MQ_INSTALLATION_PATH* représente le répertoire de haut niveau dans lequel IBM MQ est installé.

ULW *Modification de l'emplacement du référentiel de clés pour un gestionnaire de files d'attente sous UNIX, Linux, and Windows*

Vous pouvez modifier l'emplacement du fichier de la base de données de clés de votre gestionnaire de files d'attente en utilisant divers moyens, notamment la commande MQSC ALTER QMGR.

Vous pouvez modifier l'emplacement du fichier de base de données de clés de votre gestionnaire de files d'attente à l'aide de la commande MQSC ALTER QMGR pour définir l'attribut de référentiel de clés de votre gestionnaire de files d'attente. Par exemple, sous UNIX and Linux:

```
ALTER QMGR SSLKEYR('/var/mqm/qmgrs/QM1/ssl/MyKey')
```

Le fichier de la base de données de clés possède le nom de fichier qualifié complet: `/var/mqm/qmgrs/QM1/ssl/MyKey.kdb`

Sous Windows :

```
ALTER QMGR SSLKEYR('C:\Program Files\IBM\MQ\Qmgrs\QM1\ssl\Mykey')
```

Le fichier de la base de données de clés possède le nom de fichier qualifié complet: `C:\Program Files\IBM\MQ\Qmgrs\QM1\ssl\Mykey.kdb`

Avertissement : Veillez à ne pas inclure l'extension `.kdb` dans le nom de fichier du mot clé SSLKEYR, car le gestionnaire de files d'attente l'ajoute automatiquement.

Vous pouvez également modifier les attributs de votre gestionnaire de files d'attente à l'aide des commandes IBM MQ Explorer ou PCF.

Lorsque vous modifiez l'emplacement du fichier de base de données de clés d'un gestionnaire de files d'attente, les certificats ne sont pas transférés à partir de l'ancien emplacement. Si le fichier de base de données de clés auquel vous accédez est un nouveau fichier de base de données de clés, vous devez le remplir avec les certificats de l'autorité de certification et les certificats personnels dont vous avez besoin, comme décrit dans «Importation d'un certificat personnel dans un référentiel de clés sous UNIX, Linux, and Windows», à la page 316.

ULW *Localisation du référentiel de clés pour un IBM MQ MQI client sur UNIX, Linux, and Windows*

L'emplacement du référentiel de clés est indiqué par la variable MQSSLKEYR ou spécifié dans l'appel MQCONNX.

Examinez la variable d'environnement MQSSLKEYR pour trouver l'emplacement du fichier de la base de données de clés pour votre IBM MQ MQI client. Exemple :

```
echo $MQSSLKEYR
```

Vérifiez également votre application, car le nom de fichier de la base de données de clés peut également être défini dans un appel MQCONNX, comme décrit dans«Spécification de l'emplacement du référentiel de clés pour un IBM MQ MQI client sous UNIX, Linux, and Windows», à la page 301. La valeur définie dans un appel MQCONNX remplace la valeur de MQSSLKEYR.

ULW *Spécification de l'emplacement du référentiel de clés pour un IBM MQ MQI client sous UNIX, Linux, and Windows*

Il n'existe pas de référentiel de clés par défaut pour un IBM MQ MQI client. Vous pouvez spécifier son emplacement de deux manières. Assurez-vous que le fichier de la base de données de clés est accessible uniquement par les utilisateurs ou les administrateurs prévus afin d'empêcher toute copie non autorisée vers d'autres systèmes.

Vous pouvez spécifier l'emplacement du fichier de base de données de clés pour votre IBM MQ MQI client de deux manières:

- Définition de la variable d'environnement MQSSLKEYR. Par exemple, sous UNIX and Linux:

```
export MQSSLKEYR=/var/mqm/ssl/key
```

Le fichier de la base de données de clés possède le nom de fichier complet:

```
/var/mqm/ssl/key.kdb
```

Sous Windows :

```
set MQSSLKEYR=C:\Program Files\IBM\MQ\ssl\key
```

Le fichier de la base de données de clés possède le nom de fichier complet:

```
C:\Program Files\IBM\MQ\ssl\key.kdb
```

Remarque : L'extension `.kdb` est une partie obligatoire du nom de fichier, mais elle n'est pas incluse dans la valeur de la variable d'environnement.

- Indiquez le chemin et le nom de la racine du fichier de la base de données de clés dans la zone *KeyRepository* de la structure MQSCO lorsqu'une application effectue un appel MQCONNX. Pour plus d'informations sur l'utilisation de la structure MQSCO dans MQCONNX, voir Présentation de MQSCO.

ULW *Lorsque les modifications apportées aux certificats ou au magasin de certificats prennent effet sur UNIX, Linux, and Windows*

Lorsque vous modifiez les certificats dans un magasin de certificats ou l'emplacement du magasin de certificats, les modifications sont prises en compte en fonction du type de canal et de la manière dont le canal est en cours d'exécution.

Les modifications apportées aux certificats dans le fichier de base de données de clés et à l'attribut de référentiel de clés prennent effet dans les situations suivantes:

- Lorsqu'un nouveau processus de canal unique sortant exécute pour la première fois un canal TLS.
- Lorsqu'un nouveau processus de canal unique TCP/IP entrant reçoit pour la première fois une demande de démarrage d'un canal TLS.
- Lorsque la commande MQSC REFRESH SECURITY TYPE (SSL) est émise pour actualiser l'environnement TLS.
- Pour les processus d'application client, lorsque la dernière connexion TLS du processus est fermée. La prochaine connexion TLS prendra en compte les modifications apportées au certificat.
- Pour les canaux qui s'exécutent en tant qu'unités d'exécution d'un processus de regroupement de processus (amqrmppa), lorsque le processus de regroupement de processus est démarré ou redémarré et exécute d'abord un canal TLS. Si le processus de regroupement de processus a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).
- Pour les canaux qui s'exécutent en tant qu'unités d'exécution de l'initiateur de canal, lorsque l'initiateur de canal est démarré ou redémarré et qu'il exécute d'abord un canal TLS. Si le processus initiateur de canal a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).
- Pour les canaux qui s'exécutent en tant qu'unités d'exécution d'un programme d'écoute TCP/IP, lorsque le programme d'écoute est démarré ou redémarré et qu'il reçoit d'abord une demande de démarrage d'un canal TLS. Si le programme d'écoute a déjà exécuté un canal TLS et que vous souhaitez que la modification soit prise en compte immédiatement, exécutez la commande MQSC REFRESH SECURITY TYPE (SSL).

Vous pouvez également actualiser l'environnement TLS IBM MQ à l'aide des commandes IBM MQ Explorer ou PCF.

ULW *Création d'un certificat personnel autosigné sur UNIX, Linux, and Windows*

Vous pouvez créer un certificat autosigné à l'aide de **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide de **runmqckm** (iKeycmd) ou de **runmqakm** (GSKCapiCmd).

Remarque : IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .

Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.

Pour plus d'informations sur la raison pour laquelle vous pouvez utiliser des certificats autosignés, voir Utilisation de certificats autosignés pour l'authentification mutuelle de deux gestionnaires de files d'attente.

Tous les certificats numériques ne peuvent pas être utilisés avec tous les CipherSpecs. Veillez à créer un certificat compatible avec les CipherSpecs que vous devez utiliser. IBM MQ prend en charge trois types différents de CipherSpec. Pour plus de détails, voir «Interopérabilité de Elliptic Curve et de RSA CipherSpecs», à la page 46 dans la rubrique «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45 .

Pour utiliser les CipherSpecs de type 1 (dont les noms commencent par ECDHE_ECDSA_), vous devez utiliser la commande **runmqakm** pour créer le certificat et spécifier un paramètre d'algorithme de signature Elliptic Curve ECDSA ; par exemple, **-sig_alg** EC_ecdsa_with_SHA384.

Pour obtenir la liste des options disponibles avec l'algorithme de hachage **-sig_alg** , voir «Options runmqckm et runmqakm sous UNIX, Linux, and Windows», à la page 545 .

Si vous utilisez:

- Interface graphique, voir «Utilisation de l'interface utilisateur strmqikm», à la page 303
- Ligne de commande, voir «Utilisation de la ligne de commande», à la page 304

ULW *Utilisation de l'interface utilisateur **strmqikm***

Vous pouvez créer un certificat personnel à l'aide de **strmqikm** (iKeyman) Interface graphique.

Pourquoi et quand exécuter cette tâche

strmqikm ne fournit pas d'option compatible FIPS. Si vous devez gérer les certificats TLS d'une manière compatible FIPS, utilisez la commande **runmqakm** .

Procédure

Procédez comme suit pour créer un certificat personnel pour votre gestionnaire de files d'attente ou IBM MQ MQI client à l'aide de l'interface graphique:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** .
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**.
   La fenêtre **Ouvrir** s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez générer la demande ; par exemple, key . kdb.
6. Cliquez sur **OK**.
   La fenêtre **Invite de mot de passe** s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**.
   Le nom de votre fichier de base de données de clés est affiché dans la zone **Nom de fichier** .
8. Dans le menu **Créer** , cliquez sur **Nouveau certificat autosigné**. La fenêtre Créer un certificat autosigné s'affiche.

9. Dans la zone **Key Label** , entrez le libellé du certificat.

   Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID utilisateur de connexion IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.
10. Entrez ou sélectionnez une valeur pour n'importe quelle zone de la zone **Nom distinctif** ou pour l'une des zones **Nom alternatif du sujet** .
11. Pour les autres zones, acceptez le valeurs par défaut proposées ou bien tapez ou sélectionnez-en de nouvelles.

    Pour plus d'informations sur les noms distinctifs, voir «Noms distinctifs», à la page 11.
12. Cliquez sur **OK**.

    La liste **Certificats personnels** affiche le libellé du certificat personnel autosigné que vous avez créé.

Que faire ensuite

Soumettez une demande de certificat à une autorité de certification. Pour plus d'informations, voir «Réception de certificats personnels dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 310.

ULW *Utilisation de la ligne de commande*

Vous pouvez créer un certificat personnel à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd). Si vous devez gérer les certificats SSL ou TLS d'une manière compatible avec FIPS, utilisez la commande **runmqakm** .

Procédure

Créez un certificat personnel autosigné à l'aide de la commande **runmqckm** ou **runmqakm** (GSKCapiCmd).

- A l'aide de **runmqckm** sous UNIX, Linux, and Windows:

```
runmqckm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days
 -sig_alg algorithm
```

  A la place de `-dn` *distinguished_name*, vous pouvez utiliser `-san_dnsname` *DNS_names*, `-san_emailaddr` *email_addresses*ou `-san_ipaddr` *IP_addresses*.

- A l'aide de **runmqakm**:

```
runmqakm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days

        -fips -sig_alg algorithm
```

où :

-db *nom_fichier*
Indique le nom de fichier qualifié complet d'une base de données de clés CMS.

-pw *mot_de_passe*
Indique le mot de passe de la base de données de clés CMS.

-label *Libellé*
Indique le libellé de clé associé au certificat. Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` à laquelle est ajouté le nom du gestionnaire de files d'attente ou l'ID utilisateur de connexion IBM MQ MQI client , le tout en minuscules. Pour plus de détails, voir «Labels de certificat numérique, compréhension des exigences», à la page 26.

-dn *nom_distinctif*
Indique le nom distinctif X.500 entre guillemets. Au moins un attribut est requis. Vous pouvez fournir plusieurs attributs d'unité organisationnelle et de centre de données.

Remarque : Les outils **runmqckm** et **runmqakm** font référence à l'attribut de code postal POSTALCODE, et non PC. Spécifiez toujours POSTALCODE dans le paramètre **-dn** lorsque vous utilisez ces commandes de gestion de certificats pour demander des certificats avec un code postal.

-size *taille_clé*
Indique la taille de la clé. Si vous utilisez **runmqckm**, la valeur peut être 512 ou 1024. Si vous utilisez **runmqakm**, la valeur peut être 512, 1024ou 2048.

x509version *version*
Version du certificat X.509 à créer. La valeur peut être 1, 2 ou 3. La valeur par défaut est 3.

-file *nom fichier*
Indique le nom de fichier de la demande de certificat.

-expire *Jours*
Délai d'expiration en jours du certificat. La valeur par défaut est 365 jours pour un certificat.

-fips
indique que la commande est exécutée en mode FIPS. Seul le composant FIPS ICC est utilisé et ce composant doit être correctement initialisé en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqaqm** échoue.

-sig_alg
Pour **runmqckm**, indique l'algorithme de signature asymétrique utilisé pour la création de la paire de clés de l'entrée. La valeur peut être MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. La valeur par défaut est SHA1WithRSA.

-sig_alg
Pour **runmqakm**, indique l'algorithme de hachage utilisé lors de la création d'une demande de certificat. Cet algorithme de hachage est utilisé pour créer la signature associée à la demande de certificat nouvellement créée. La valeur peut être md5, MD5_WITH_RSA, MD5WithRSA, SHA_WITH_DSA, SHA_WITH_RSA, sha1, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, sha224, SHA224_WITH_RSA, SHA224WithDSA, SHA224WithECDSA, SHA224WithRSA, sha256, SHA256_WITH_RSA, SHA256WithDSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithRSA, sha384, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, sha512, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHAWithDSA, SHAWithRSA, EC_ecdsa_with_SHA1, EC_ecdsa_with_SHA224, EC_ecdsa_with_SHA256, EC_ecdsa_with_SHA384ou EC_ecdsa_with_SHA512. La valeur par défaut est SHA1WithRSA.

-san_dnsname *noms_DNS*
Indique une liste de noms DNS séparés par des virgules ou des espaces pour l'entrée en cours de création.

-san_emailaddr *adresse_e-mail*
Indique une liste d'adresses électroniques séparées par des virgules ou des espaces pour l'entrée en cours de création.

-san_ipaddr *Adresse_IP*
Indique une liste d'adresses IP séparées par des virgules ou des espaces pour l'entrée en cours de création.

Que faire ensuite

Soumettez une demande de certificat à une autorité de certification. Pour plus d'informations, voir «Réception de certificats personnels dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 310.

ULW *Demande d'un certificat personnel sur UNIX, Linux, and Windows*

Vous pouvez demander un certificat personnel à l'aide du **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd). Si vous devez gérer les certificats SSL ou TLS d'une manière compatible avec FIPS, utilisez la commande **runmqakm** .

Pourquoi et quand exécuter cette tâche

Vous pouvez demander un certificat personnel à l'aide de l'interface graphique d' **strmqikm** ou à partir de la ligne de commande, sous réserve des remarques suivantes:

- IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .
- Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.
- Tous les certificats numériques ne peuvent pas être utilisés avec tous les CipherSpecs. Veillez à demander un certificat compatible avec les CipherSpecs que vous devez utiliser. IBM MQ prend en charge trois types différents de CipherSpec. Pour plus de détails, voir «Interopérabilité de Elliptic Curve et de RSA CipherSpecs», à la page 46 dans la rubrique «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45 .
- Pour utiliser les CipherSpecs de type 1 (dont les noms commencent par ECDHE_ECDSA_), vous devez utiliser la commande **runmqakm** pour demander le certificat et spécifier un paramètre d'algorithme de signature Elliptic Curve ECDSA ; par exemple, **-sig_alg** `EC_ecdsa_with_SHA384`.

  Pour obtenir la liste des options disponibles avec l'algorithme de hachage **-sig_alg** , voir «Options runmqckm et runmqakm sous UNIX, Linux, and Windows», à la page 545 .
- Seule la commande **runmqakm** fournit une option compatible FIPS.
- Si vous utilisez du matériel de cryptographie, voir «Demande d'un certificat personnel pour votre matériel PKCS #11», à la page 325.

Si vous utilisez:

- Interface graphique, voir «Utilisation de l'interface utilisateur strmqikm», à la page 306
- Ligne de commande, voir «Utilisation de la ligne de commande», à la page 307

ULW *Utilisation de l'interface utilisateur **strmqikm***

Vous pouvez demander un certificat personnel à l'aide du **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd). Si vous devez gérer les certificats SSL ou TLS d'une manière compatible avec FIPS, utilisez la commande **runmqakm** .

Pourquoi et quand exécuter cette tâche

strmqikm ne fournit pas d'option compatible FIPS. Si vous devez gérer les certificats TLS d'une manière compatible FIPS, utilisez la commande **runmqakm** .

Procédure

Pour appliquer un certificat personnel à l'aide de l'interface utilisateur iKeyman , procédez comme suit:

1. Démarrez l'interface utilisateur à l'aide de la commande **strmqikm** .

2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**.

   La fenêtre **Ouvrir** s'ouvre.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez générer la demande ; par exemple, `key.kdb`.
6. Cliquez sur **Ouvrir**.

   La fenêtre **Invite de mot de passe** s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**.

   Le nom de votre fichier de base de données de clés est affiché dans la zone **Nom de fichier** .
8. Dans le menu **Créer** , cliquez sur **Nouvelle demande de certificat**. La fenêtre **Créer une demande de clé et de certificat** s'ouvre.
9. Dans la zone **Key Label** , entrez le libellé du certificat.

   Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID utilisateur de connexion IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.
10. Entrez ou sélectionnez une valeur pour n'importe quelle zone de la zone **Nom distinctif** ou pour l'une des zones **Nom alternatif du sujet** . Pour les autres zones, acceptez le valeurs par défaut proposées ou bien tapez ou sélectionnez-en de nouvelles.

    Pour plus d'informations sur les noms distinctifs, voir «Noms distinctifs», à la page 11.
11. Dans la zone **Entrez le nom d'un fichier dans lequel stocker la demande de certificat** , acceptez la valeur par défaut `certreq.arm`ou entrez une nouvelle valeur avec un chemin d'accès complet.
12. Cliquez sur **OK**.

    Une fenêtre de confirmation s'affiche.
13. Cliquez sur **OK**.

    La liste **Demandes de certificat personnel** affiche le libellé de la nouvelle demande de certificat personnel que vous avez créée. La demande de certificat est stockée dans le fichier que vous avez choisi à l'étape «11», à la page 307.
14. Demandez le nouveau certificat personnel soit en envoyant le fichier à une autorité de certification, soit en copiant le fichier dans le formulaire de demande sur le site Web de l'autorité de certification.

ULW *Utilisation de la ligne de commande*

Vous pouvez demander un certificat personnel à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd). Si vous devez gérer les certificats SSL ou TLS d'une manière compatible avec FIPS, utilisez la commande **runmqakm** .

Procédure

Demandez un certificat personnel à l'aide de la commande **runmqckm** ou **runmqakm** (GSKCapiCmd).

- A l'aide de **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
         -dn distinguished_name -size key_size
        -file filename -sig_alg algorithm
```

  A la place de `-dn distinguished_name`, vous pouvez utiliser `-san_dsname DNS_names`, `-san_emailaddr email_addresses`ou `-san_ipaddr IP_addresses`.

- A l'aide de **runmqakm**:

```
runmqakm -certreq -create -db filename -pw
password -label label
```

```
          -dn distinguished_name -size key_size
          -file filename -fips -sig_alg algorithm
```

où :

-db *nom_fichier*
Indique le nom de fichier qualifié complet d'une base de données de clés CMS.

-pw *mot_de_passe*
Indique le mot de passe de la base de données de clés CMS.

-label *Libellé*
Indique le libellé de clé associé au certificat. Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` à laquelle est ajouté le nom du gestionnaire de files d'attente ou l'ID utilisateur de connexion IBM MQ MQI client , le tout en minuscules. Pour plus de détails, voir «Labels de certificat numérique, compréhension des exigences», à la page 26.

-dn *nom_distinctif*
Indique le nom distinctif X.500 entre guillemets. Au moins un attribut est requis. Vous pouvez fournir plusieurs attributs d'unité organisationnelle et de centre de données.

Remarque : Les outils **runmqckm** et **runmqakm** font référence à l'attribut de code postal `POSTALCODE`, et non PC. Spécifiez toujours `POSTALCODE` dans le paramètre **-dn** lorsque vous utilisez ces commandes de gestion de certificats pour demander des certificats avec un code postal.

-size *taille_clé*
Indique la taille de la clé. Si vous utilisez **runmqckm**, la valeur peut être 512 ou 1024. Si vous utilisez **runmqakm**, la valeur peut être 512, 1024ou 2048.

-file *nom fichier*
Indique le nom de fichier de la demande de certificat.

-fips
indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

-sig_alg
Pour **runmqckm**, indique l'algorithme de signature asymétrique utilisé pour la création de la paire de clés de l'entrée. La valeur peut être `MD2_WITH_RSA`, `MD2WithRSA`, `MD5_WITH_RSA`, `MD5WithRSA`, `SHA1WithDSA`, `SHA1WithECDSA`, `SHA1WithRSA`, `SHA2/ECDSA`, `SHA224WithECDSA`, `SHA256_WITH_RSA`, `SHA256WithECDSA`, `SHA256WithRSA`, `SHA2WithECDSA`, `SHA3/ECDSA`, `SHA384_WITH_RSA`, `SHA384WithECDSA`, `SHA384WithRSA`, `SHA3WithECDSA`, `SHA5/ECDSA`, `SHA512_WITH_RSA`, `SHA512WithECDSA`, `SHA512WithRSA`, `SHA5WithECDSA`, `SHA_WITH_DSA`, `SHA_WITH_RSA`, `SHAWithDSA`, `SHAWithRSA`. La valeur par défaut est `SHA1WithRSA`.

-sig_alg
Pour **runmqakm**, indique l'algorithme de hachage utilisé lors de la création d'une demande de certificat. Cet algorithme de hachage est utilisé pour créer la signature associée à la demande de certificat nouvellement créée. La valeur peut être `md5`, `MD5_WITH_RSA`, `MD5WithRSA`, `SHA_WITH_DSA`, `SHA_WITH_RSA`, `sha1`, `SHA1WithDSA`, `SHA1WithECDSA`, `SHA1WithRSA`, `sha224`, `SHA224_WITH_RSA`, `SHA224WithDSA`, `SHA224WithECDSA`, `SHA224WithRSA`, `sha256`, `SHA256_WITH_RSA`, `SHA256WithDSA`, `SHA256WithECDSA`, `SHA256WithRSA`, `SHA2WithRSA`, `sha384`, `SHA384_WITH_RSA`, `SHA384WithECDSA`, `SHA384WithRSA`, `sha512`, `SHA512_WITH_RSA`, `SHA512WithECDSA`, `SHA512WithRSA`, `SHAWithDSA`, `SHAWithRSA`, `EC_ecdsa_with_SHA1`, `EC_ecdsa_with_SHA224`, `EC_ecdsa_with_SHA256`, `EC_ecdsa_with_SHA384`ou `EC_ecdsa_with_SHA512`. La valeur par défaut est `SHA1WithRSA`.

-san_dnsname *noms_DNS*
Indique une liste de noms DNS séparés par des virgules ou des espaces pour l'entrée en cours de création.

-san_emailaddr *adresse_e-mail*
Indique une liste d'adresses électroniques séparées par des virgules ou des espaces pour l'entrée en cours de création.

-san_ipaddr ***Adresse_IP***
Indique une liste d'adresses IP séparées par des virgules ou des espaces pour l'entrée en cours de création.

Que faire ensuite

Soumettez une demande de certificat à une autorité de certification. Pour plus d'informations, voir «Réception de certificats personnels dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 310.

ULW *Renouvellement d'un certificat personnel existant sous UNIX, Linux, and Windows*

Vous pouvez renouveler un certificat personnel à l'aide du **strmqikm** (iKeyman) Interface graphique ou à partir de la ligne de commande à l'aide des commandes **runmqckm** (iKeycmd) ou **runmqakm** (GSKCapiCmd).

Pourquoi et quand exécuter cette tâche

Si vous devez utiliser des tailles de clé plus grandes pour vos certificats personnels, vous ne pouvez pas renouveler un certificat existant. Vous devez remplacer votre clé existante en suivant les étapes décrites dans «Demande d'un certificat personnel sur UNIX, Linux, and Windows», à la page 306 pour créer une nouvelle demande de certificat qui utilise les tailles de clé dont vous avez besoin.

Un certificat personnel a une date d'expiration, après laquelle le certificat ne peut plus être utilisé. Cette tâche explique comment renouveler un certificat personnel existant avant son expiration.

Utilisation de l'interface utilisateur ***strmqikm***

Pourquoi et quand exécuter cette tâche

strmqikm ne fournit pas d'option compatible FIPS. Si vous devez gérer les certificats TLS d'une manière compatible FIPS, utilisez la commande **runmqakm** .

Procédure

Pour appliquer un certificat personnel à l'aide de l'interface utilisateur **strmqikm** , procédez comme suit:

1. Démarrez l'interface utilisateur à l'aide de la commande **strmqikm** sous UNIX, Linux, and Windows.
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**.
   La fenêtre **Ouvrir** s'ouvre.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez générer la demande ; par exemple, `key.kdb`.
6. Cliquez sur **Ouvrir**.
   La fenêtre **Invite de mot de passe** s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**.
   Le nom de votre fichier de base de données de clés est affiché dans la zone **Nom de fichier** .
8. Sélectionnez **Personal Certificates** dans le menu déroulant de sélection, puis sélectionnez le certificat à renouveler dans la liste.
9. Cliquez sur **Re-create Request ...** bouton.
   Une fenêtre s'ouvre pour que vous puissiez entrer le nom de fichier et les informations d'emplacement de fichier.
10. Dans la zone **nom de fichier** , acceptez la valeur par défaut `certreq.arm`ou entrez une nouvelle valeur, y compris le chemin d'accès complet au fichier.

11. Cliquez sur **OK**. La demande de certificat est stockée dans le fichier que vous avez sélectionné à l'étape «9», à la page 309.
12. Demandez le nouveau certificat personnel soit en envoyant le fichier à une autorité de certification, soit en copiant le fichier dans le formulaire de demande sur le site Web de l'autorité de certification.

Utilisation de la ligne de commande

Procédure

Utilisez les commandes suivantes pour demander un certificat personnel à l'aide de la commande **runmqckm** ou **runmqakm** :

- Utilisation de **runmqckm** sur les systèmes UNIX, Linux, and Windows :

```
runmqckm -certreq -recreate -db filename -pw
password -label label
-target filename
```

- Utilisation de runmqakm:

```
runmqakm -certreq -recreate -db filename -pw
password -label label
-target filename
```

où :

-db *nom_fichier*
Indique le nom de fichier qualifié complet d'une base de données de clés CMS.

-pw *mot_de_passe*
Indique le mot de passe de la base de données de clés CMS.

-target *nom_fichier*
Indique le nom de fichier de la demande de certificat.

Que faire ensuite

Une fois que vous avez reçu le certificat personnel signé de l'autorité de certification, vous pouvez l'ajouter à votre base de données de clés en suivant les étapes décrites dans «Réception de certificats personnels dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 310.

ULW *Réception de certificats personnels dans un référentiel de clés sur UNIX, Linux, and Windows*

Utilisez cette procédure pour recevoir un certificat personnel dans le fichier de la base de données de clés. Le référentiel de clés doit être le même que celui dans lequel vous avez créé la demande de certificat.

Une fois que l'autorité de certification vous a envoyé un nouveau certificat personnel, vous l'ajoutez au fichier de base de données de clés à partir duquel vous avez généré la nouvelle demande de certificat. Si l'autorité de certification envoie le certificat dans le cadre d'un message électronique, copiez le certificat dans un fichier distinct.

Utilisation strmqikm

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** . **strmqikm** ne fournit pas d'option compatible FIPS.

Vérifiez que le fichier certificat à importer dispose des droits d'accès en écriture pour l'utilisateur en cours, puis utilisez la procédure suivante pour un gestionnaire de files d'attente ou un IBM MQ MQI client afin de recevoir un certificat personnel dans le fichier de la base de données de clés:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** (sous Windows UNIX and Linux).

2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de clés dans lequel vous souhaitez ajouter le certificat, par exemple `key.kdb`.
6. Cliquez sur **Ouvrir**, puis sur **OK**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de base de données de clés s'affiche dans la zone **Nom de fichier** . Sélectionnez la vue **Certificats personnels** .
8. Cliquez sur **Receive**. La fenêtre Recevoir un certificat d'un fichier s'affiche.
9. Entrez le nom du fichier de certificat et l'emplacement du nouveau certificat personnel ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
10. Cliquez sur **OK**. Si vous disposez déjà d'un certificat personnel dans votre base de données de clés, une fenêtre s'ouvre pour vous demander si vous souhaitez définir la clé que vous ajoutez comme clé par défaut dans la base de données.
11. Cliquez sur **Oui** ou sur **Non**. La fenêtre Enter a Label s'ouvre.
12. Cliquez sur **OK**. La zone **Certificats personnels** affiche le libellé du nouveau certificat personnel que vous avez ajouté.

Utilisation de la ligne de commande

Pour ajouter un certificat personnel à un fichier de base de données de clés, utilisez l'une des commandes suivantes:

- A l'aide de **runmqckm**:

```
runmqckm -cert -receive -file filename -db filename -pw password
         -format ascii
```

- A l'aide de **runmqakm**:

```
runmqakm -cert -receive -file filename -db filename -pw password -fips
```

où :

-file *nom fichier*
Indique le nom de fichier qualifié complet du certificat personnel.

-db *nom_fichier*
Indique le nom de fichier qualifié complet d'une base de données de clés CMS.

-pw *mot_de_passe*
Indique le mot de passe de la base de données de clés CMS.

-format *ascii*
Indique le format du certificat. La valeur peut-être `ascii` pour les données ASCII codées en base 64 ou `binary` pour les données Binary DER. La valeur par défaut est `ascii`.

-fips
indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

Si vous utilisez du matériel de cryptographie, voir «Réception d'un certificat personnel dans votre matériel PKCS #11», à la page 326.

ULW *Extraction d'un certificat de l'autorité de certification à partir d'un référentiel de clés sur UNIX, Linux, and Windows*

Procédez comme suit pour extraire un certificat d'autorité de certification.

Utilisation `strmqikm`

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** . **strmqikm** (iKeyman) ne fournit pas d'option compatible FIPS.

Effectuez les étapes suivantes sur la machine à partir de laquelle vous souhaitez extraire le certificat de l'autorité de certification:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** .
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez extraire, par exemple `key.kdb`.
6. Cliquez sur **Ouvrir**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de base de données de clés s'affiche dans la zone **Nom de fichier** .
8. Dans la zone **Contenu de la base de données de clés** , sélectionnez **Certificats de signataire** et sélectionnez le certificat à extraire.
9. Cliquez sur **Extraire**. La fenêtre Extraire un certificat dans un fichier s'ouvre.
10. Sélectionnez le **Type de données** du certificat, par exemple **Base64-encoded codées en Base64** pour un fichier avec l'extension `.arm` .
11. Entrez le nom du fichier de certificat et l'emplacement où vous souhaitez stocker le certificat ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
12. Cliquez sur **OK**. Le certificat est écrit dans le fichier que vous avez spécifié.

Utilisation de la ligne de commande

Utilisez les commandes suivantes pour extraire un certificat d'autorité de certification à l'aide de **runmqckm**:

- Sous UNIX, Linux, and Windows :

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

où :

`-db` *`filename`*	est le nom de chemin qualifié complet d'une base de données de clés CMS.
`-pw` *`password`*	correspond au mot de passe de la base de données de clés CMS.
`-label` *`label`*	correspond au libellé du certificat.
`-target` *`filename`*	est le nom du fichier de destination.
`-format` *`ascii`*	correspond au format du certificat. La valeur peut-être `ascii` pour les données ASCII codées en base 64 ou `binary` pour les données Binary DER. La valeur par défaut est `ascii`.
`-fips`	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

ULW *Extraction de la partie publique d'un certificat autosigné à partir d'un référentiel de clés sur UNIX, Linux, and Windows*

Procédez comme suit pour extraire la partie publique d'un certificat autosigné.

Utilisation `strmqikm`

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** . **strmqikm** (iKeyman) ne fournit pas d'option compatible FIPS.

Effectuez les étapes suivantes sur la machine à partir de laquelle vous souhaitez extraire la partie publique d'un certificat autosigné:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** (sous UNIX, Linux, and Windows).
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez extraire le certificat, par exemple `key.kdb`.
6. Cliquez sur **OK**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de base de données de clés s'affiche dans la zone **Nom de fichier** .
8. Dans la zone **Contenu de la base de données de clés** , sélectionnez **Certificats personnels** et sélectionnez le certificat.
9. Cliquez sur **Extraire le certificat**. La fenêtre Extraire un certificat dans un fichier s'ouvre.
10. Sélectionnez le **Type de données** du certificat, par exemple **Base64-encoded codées en Base64** pour un fichier avec l'extension `.arm` .
11. Entrez le nom du fichier de certificat et l'emplacement où vous souhaitez stocker le certificat ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
12. Cliquez sur **OK**. Le certificat est écrit dans le fichier que vous avez spécifié. Notez que lorsque vous extrayez (plutôt que d'exporter) un certificat, seule la partie publique du certificat est incluse, de sorte qu'un mot de passe n'est pas requis.

Utilisation de la ligne de commande

Utilisez les commandes suivantes pour extraire la partie publique d'un certificat autosigné à l'aide de **runmqckm** ou de **runmqakm**:

- Sous UNIX, Linux, and Windows :

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

- Utilisation de runmqakm:

```
runmqakm -cert -extract -db filename -pw password -label label
            -target filename -format ascii -fips
```

où :

`-db` *filename*	est le nom de chemin qualifié complet d'une base de données de clés CMS.
`-pw` *password*	correspond au mot de passe de la base de données de clés CMS.
`-label` *label*	correspond au libellé du certificat.

`-target` *`filename`*	est le nom du fichier de destination.
`-format` *`ascii`*	correspond au format du certificat. La valeur peut-être `ascii` pour les données ASCII codées en base 64 ou `binary` pour les données Binary DER. La valeur par défaut est `ascii`.
`-fips`	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **`runmqakm`** échoue.

ULW *Ajout d'un certificat de l'autorité de certification ou de la partie publique d'un certificat autosigné dans un référentiel de clés sur UNIX, Linux, and Windows*

Suivez cette procédure pour ajouter un certificat de l'autorité de certification ou la partie publique d'un certificat autosigné au référentiel principal.

Si le certificat que vous souhaitez ajouter se trouve dans une chaîne de certificats, vous devez également ajouter tous les certificats se trouvant au-dessus de ce dernier dans la chaîne. Vous devez absolument ajouter les certificats dans l'ordre décroissant en commençant par la racine, puis par le certificat de l'autorité de certification situé immédiatement en-dessous dans la chaîne, etc.

Lorsque les instructions suivantes font référence à un certificat de l'autorité de certification, elles s'appliquent également à la partie publique d'un certificat autosigné.

Remarque : Vous devez vous assurer que le certificat est en codage ASCII (UTF-8) ou binaire (DER), car IBM Global Secure Toolkit (GSKit) ne prend pas en charge les certificats avec d'autres types de codage.

Utilisation `strmqikm`

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **`runmqakm`** . **`strmqikm`** ne fournit pas d'option compatible FIPS.

Suivez les étapes suivantes sur la machine sur laquelle vous souhaitez ajouter le certificat de l'autorité de certification :

1. Démarrez l'interface graphique à l'aide de la commande **`strmqikm`** (sur les systèmes UNIX, Linux et Windows).
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de clés dans lequel vous souhaitez ajouter le certificat, par exemple `key.kdb`.
6. Cliquez sur **OK**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de clés s'affiche dans la zone **Nom du fichier**.
8. Dans la zone **Key database content**, sélectionnez **Signer Certificates**.
9. Cliquez sur **Ajouter**. La fenêtre Add CA's Certificate from a File s'ouvre.
10. Entrez le nom et l'emplacement du fichier certificat ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
11. Cliquez sur **OK**. La fenêtre Enter a Label s'ouvre.
12. Dans la fenêtre Enter a Label, entrez le nom du certificat.
13. Cliquez sur **OK**. Le certificat est ajouté à la base de données de clés.

Utilisation de la ligne de commande

Pour ajouter un certificat d'autorité de certification à une base de données de clés, utilisez l'une des commandes suivantes:

- A l'aide de **runmqckm**:

```
runmqckm -cert -add -db filename -pw password -label label
         -file filename -format ascii
```

- A l'aide de **runmqakm**:

```
runmqakm -cert -add -db filename -pw password -label label
         -file filename -format ascii -fips
```

où :

-db *nom_fichier*
Indique le nom de fichier qualifié complet de la base de données de clés CMS.

-pw *mot_de_passe*
Indique le mot de passe de la base de données de clés CMS.

-label *Libellé*
Indique le libellé associé au certificat.

-file *nom fichier*
Indique le nom du fichier contenant le certificat.

-format *ascii*
Indique le format du certificat. La valeur peut-être `ascii` pour les données ASCII codées en base 64 ou `binary` pour les données Binary DER. La valeur par défaut est `ascii`.

-fips
indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

ULW *Exportation d'un certificat personnel à partir d'un référentiel de clés sous UNIX, Linux, and Windows*

Suivez cette procédure pour exporter un certificat personnel.

Utilisation `strmqikm`

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** . **strmqikm** (iKeyman) ne fournit pas d'option compatible FIPS.

Effectuez les étapes suivantes sur la machine à partir de laquelle vous souhaitez exporter le certificat personnel:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** (sous Windows UNIX and Linux).
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez exporter le certificat, par exemple `key.kdb`.
6. Cliquez sur **Ouvrir**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de base de données de clés s'affiche dans la zone **Nom de fichier** .
8. Dans la zone **Contenu de la base de données de clés** , sélectionnez **Certificats personnels** et sélectionnez le certificat à exporter.
9. Cliquez sur **Exporter / Importer**. La fenêtre Exportation / Importation de clé s'ouvre.
10. Sélectionnez **Exporter la clé**.
11. Sélectionnez le **Type de fichier de clés** du certificat à exporter, par exemple **PKCS12**.

12. Entrez le nom de fichier et l'emplacement vers lesquels vous souhaitez exporter le certificat ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
13. Cliquez sur **OK**. La fenêtre Password Prompt s'ouvre. Notez que lorsque vous exportez (plutôt que d'extraire) un certificat, les parties publique et privée du certificat sont incluses. C'est pourquoi le fichier exporté est protégé par un mot de passe. Lorsque vous extrayez un certificat, seule la partie publique du certificat est incluse, de sorte qu'un mot de passe n'est pas requis.
14. Entrez un mot de passe dans la zone **Mot de passe** , puis entrez-le à nouveau dans la zone **Confirmer le mot de passe** .
15. Cliquez sur **OK**. Le certificat est exporté dans le fichier que vous avez spécifié.

Utilisation de la ligne de commande

Utilisez les commandes suivantes pour exporter un certificat personnel à l'aide de **runmqckm**:

- Sous UNIX, Linux, and Windows :

```
runmqckm -cert -export -db filename -pw password -label label -type cms
        -target filename -target_pw password -target_type pkcs12
```

où :

-db *filename*	correspond au nom de chemin qualifié complet de la base de données de clés CMS.
-fips	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.
-pw *password*	correspond au mot de passe de la base de données de clés CMS.
-label *label*	correspond au libellé du certificat.
-type *cms*	est le type de la base de données.
-target *filename*	est le nom de chemin qualifié complet du fichier de destination.
-target_pw *password*	est le mot de passe utilisé pour le chiffrement du certificat.
-target_type *pkcs12*	est le type du certificat.

ULW *Importation d'un certificat personnel dans un référentiel de clés sous UNIX, Linux, and Windows*

Suivez cette procédure pour importer un certificat personnel

Avant d'importer un certificat personnel au format PKCS #12 dans le fichier de base de données de clés, vous devez d'abord ajouter la chaîne valide complète d'émission de certificats de l'autorité de certification au fichier de base de données de clés (voir «Ajout d'un certificat de l'autorité de certification ou de la partie publique d'un certificat autosigné dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 314).

Les fichiers PKCS #12 doivent être considérés comme temporaires et supprimés après utilisation.

Utilisation strmqikm

Si vous devez gérer les certificats TLS d'une manière compatible FIPS, utilisez la commande **runmqakm** . **strmqikm** ne fournit pas d'option compatible FIPS.

Effectuez les étapes suivantes sur la machine sur laquelle vous souhaitez importer le certificat personnel:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** .
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.

3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de clés dans lequel vous souhaitez ajouter le certificat, par exemple `key.kdb`.
6. Cliquez sur **Ouvrir**. La fenêtre d'invite de mot de passe s'affiche.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de clés s'affiche dans la zone **Nom du fichier**.
8. Dans la zone **Contenu de la base de données de clés** , sélectionnez **Certificats personnels**.
9. Si la vue Certificats personnels contient des certificats, procédez comme suit:
   a. Cliquez sur **Exporter / Importer**. La fenêtre Export / Import key s'affiche.
   b. Sélectionnez **Importer la clé**.
10. S'il n'existe aucun certificat dans la vue Certificats personnels, cliquez sur **Importer**.
11. Sélectionnez le **Type de fichier de clés** du certificat à importer, par exemple PKCS12.
12. Entrez le nom et l'emplacement du fichier certificat ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
13. Cliquez sur **OK**. La fenêtre d'invite de mot de passe s'affiche.
14. Dans la zone **Mot de passe** , entrez le mot de passe utilisé lors de l'exportation du certificat.
15. Cliquez sur **OK**. La fenêtre Modifier les libellés s'affiche. Vous pouvez modifier les libellés des certificats importés si, par exemple, un certificat portant le même libellé existe déjà dans la base de données de clés cible. La modification des libellés de certificat n'a aucun effet sur la validation de la chaîne de certificats. Pour associer le certificat à un gestionnaire de files d'attente particulier ou à IBM MQ MQI client, IBM MQ utilise soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID de connexion utilisateur IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.
16. Pour modifier un libellé, sélectionnez-le dans la liste **Sélectionner un libellé à modifier** . Le libellé est copié dans la zone d'entrée **Entrer un nouveau libellé** . Remplacez le texte du libellé par celui du nouveau libellé et cliquez sur **Appliquer**.
17. Le texte de la zone d'entrée **Entrer un nouveau libellé** est recopié dans la zone **Sélectionner un libellé à modifier** , en remplaçant le libellé sélectionné à l'origine et en réétiquetant le certificat correspondant.
18. Une fois que vous avez modifié tous les libellés à modifier, cliquez sur **OK**. La fenêtre Modifier les libellés se ferme et la fenêtre Gestion des clés IBM d'origine réapparaît avec les zones **Certificats personnels** et **Certificats de signataire** mises à jour avec les certificats correctement libellés.
19. Le certificat est importé dans la base de données de clés cible.

Utilisation de la ligne de commande

Pour importer un certificat personnel à l'aide de **runmqckm**, utilisez la commande suivante:

- Sous UNIX, Linux, and Windows :

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label
```

où :

`-file filename`	est le nom de fichier qualifié complet du fichier contenant le certificat PKCS #12 .
`-pw password`	est le mot de passe du certificat PKCS #12 .
`-type pkcs12`	est le type du fichier.

`-target` *filename*	est le nom de la base de données de clés CMS de destination.
`-target_pw` *password*	correspond au mot de passe de la base de données de clés CMS.
`-target_type` *cms*	est le type de la base de données spécifiée par `-target`
`-label` *label*	est le libellé du certificat à importer à partir de la base de données de clés source.
`-new_label` *label*	est le libellé auquel le certificat sera affecté dans la base de données cible. Si vous omettez l'option `-new_label` , l'option `-label` est utilisée par défaut.
`-fips`	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

runmqckm ne fournit pas de commande permettant de modifier directement les libellés de certificat. Pour modifier un libellé de certificat, procédez comme suit:

1. Exportez le certificat dans un fichier PKCS #12 à l'aide de la commande **-cert -export** . Indiquez le libellé de certificat existant pour l'option `-label` .
2. Supprimez la copie existante du certificat de la base de données de clés d'origine à l'aide de la commande **-cert -delete** .
3. Importez le certificat à partir du fichier PKCS #12 à l'aide de la commande **-cert -import** . Spécifiez l'ancien libellé pour l'option `-label` et le nouveau libellé requis pour l'option `-new_label` . Le certificat sera réimporté dans la base de données de clés avec le libellé requis.

ULW *Importation d'un certificat personnel à partir d'un fichier Microsoft.pfx*

Suivez cette procédure pour effectuer une importation à partir d'un fichier Microsoft.pfx sous UNIX, Linux, and Windows.

Un fichier .pfx peut contenir deux certificats relatifs à la même clé. L'un est un certificat personnel ou de site (contenant à la fois une clé publique et une clé privée). L'autre est un certificat de l'autorité de certification (signataire) (contenant uniquement une clé publique). Ces certificats ne pouvant pas coexister dans le même fichier de base de données de clés CMS, un seul d'entre eux peut être importé. En outre, le "nom usuel" ou le libellé est joint uniquement au certificat de signataire.

Le certificat personnel est identifié par un identificateur unique (UUID) généré par le système. Cette section montre l'importation d'un certificat personnel à partir d'un fichier pfx tout en l'étiquetant avec le nom usuel précédemment affecté au certificat de l'autorité de certification (signataire). Les certificats de l'autorité de certification émettrice (signataire) doivent déjà être ajoutés à la base de données de clés cible. Notez que les fichiers PKCS#12 doivent être considérés comme temporaires et supprimés après utilisation.

Pour importer un certificat personnel à partir d'une base de données de clés pfx source, procédez comme suit:

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** . La fenêtre Gestion des clés IBM s'affiche.
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Sélectionnez le type de base de données de clés **PKCS12**.
4. **Il est recommandé d'effectuer une sauvegarde de la base de données pfx avant d'effectuer cette étape.** Sélectionnez la base de données de clés pfx à importer. Cliquez sur **Ouvrir**. La fenêtre de saisie de mot de passe s'affiche.
5. Entrez le mot de passe de la base de données de clés et cliquez sur **OK**. La fenêtre Gestion des clés IBM s'affiche. La barre de titre affiche le nom du fichier de base de données de clés pfx sélectionné, indiquant que le fichier est ouvert et prêt.

6. Sélectionnez **Signer Certificates** dans la liste. Le "nom usuel" du certificat requis s'affiche sous la forme d'un libellé dans le panneau Certificats de signataire.
7. Sélectionnez l'entrée de libellé et cliquez sur **Supprimer** pour supprimer le certificat de signataire. La fenêtre de confirmation s'affiche.
8. Cliquez sur **Oui**. Le libellé sélectionné n'est plus affiché dans le panneau Certificats de signataire.
9. Répétez les étapes 6, 7 et 8 pour tous les certificats de signataire.
10. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
11. Sélectionnez la base de données CMS de clés cible dans laquelle le fichier pfx est importé. Cliquez sur **Ouvrir**. La fenêtre de saisie de mot de passe s'affiche.
12. Entrez le mot de passe de la base de données de clés et cliquez sur **OK**. La fenêtre Gestion des clés IBM s'affiche. La barre de titre affiche le nom du fichier de base de données de clés sélectionné, indiquant que le fichier est ouvert et prêt.
13. Sélectionnez **Certificats personnels** dans la liste.
14. Si la vue Certificats personnels contient des certificats, procédez comme suit:
    a. Cliquez sur **Exporter / Importer la clé**. La fenêtre Export / Import key s'affiche.
    b. Sélectionnez **Importer** dans Choisir le type d'action.
15. S'il n'existe aucun certificat dans la vue Certificats personnels, cliquez sur **Importer**.
16. Sélectionnez le fichier PKCS12 .
17. Entrez le nom du fichier pfx tel qu'il est utilisé à l'étape 4. Cliquez sur **OK**. La fenêtre de saisie de mot de passe s'affiche.
18. Indiquez le même mot de passe que celui que vous avez indiqué lorsque vous avez supprimé le certificat de signataire. Cliquez sur **OK**.
19. La fenêtre Modifier les libellés s'affiche (car il ne doit y avoir qu'un seul certificat disponible pour l'importation). Le libellé du certificat doit être un identificateur unique universel au format `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
20. Pour modifier le libellé, sélectionnez l'identificateur unique universel dans le panneau **Sélectionner un libellé à modifier:** . Le libellé sera répliqué dans la zone **Entrez un nouveau libellé:** . Remplacez le texte du libellé par celui du nom usuel qui a été supprimé à l'étape 7 et cliquez sur **Appliquer**. Le nom usuel doit être soit la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID de connexion utilisateur IBM MQ MQI client ajouté, en minuscules. Pour plus de détails voir Labels de certificat numérique.
21. Cliquez sur **OK**. La fenêtre Modifier les libellés est maintenant supprimée et la fenêtre de gestion des clés IBM d'origine réapparaît avec les panneaux Certificats personnels et Certificats de signataire mis à jour avec le certificat personnel correctement libellé.
22. Le certificat personnel pfx est maintenant importé dans la base de données (cible).

Il n'est pas possible de modifier un libellé de certificat à l'aide de **runmqckm** ou **runmqakm**.

Utilisation de la ligne de commande

Pour importer un certificat personnel à l'aide de **runmqckm** sous UNIX, Linux, and Windows, utilisez la commande suivante:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label -pfx
```

Pour importer un certificat personnel à l'aide de **runmqakm**, utilisez la commande suivante:

```
runmqakm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label -fips -pfx
```

où :

`-file` *filename*	est le nom de fichier qualifié complet du fichier contenant le certificat PKCS #12 .
`-pw` *password*	est le mot de passe du certificat PKCS #12 .
`-type` *pkcs12*	est le type du fichier.
`-target` *filename*	est le nom de la base de données de clés CMS de destination.
`-target_pw` *password*	correspond au mot de passe de la base de données de clés CMS.
`-target_type` *cms*	est le type de la base de données spécifiée par `-target`
`-label` *label*	est le libellé du certificat à importer à partir de la base de données de clés source.
`-new_label` *label*	est le libellé auquel le certificat sera affecté dans la base de données cible. Si vous omettez l'option `-new_label` , l'option `-label` est utilisée par défaut.
`-fips`	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.
`-pfx`	indique le format de fichier PFX.

runmqckm ne fournit pas de commande permettant de modifier directement les libellés de certificat. Pour modifier un libellé de certificat, procédez comme suit:

1. Exportez le certificat dans un fichier PKCS #12 à l'aide de la commande **-cert -export** . Indiquez le libellé de certificat existant pour l'option `-label` .
2. Supprimez la copie existante du certificat de la base de données de clés d'origine à l'aide de la commande **-cert -delete** .
3. Importez le certificat à partir du fichier PKCS #12 à l'aide de la commande **-cert -import** . Spécifiez l'ancien libellé pour l'option `-label` et le nouveau libellé requis pour l'option `-new_label` . Le certificat sera réimporté dans la base de données de clés avec le libellé requis.

ULW *Importation d'un certificat personnel à partir d'un fichier PKCS #7*

Les outils **strmqikm** (iKeyman) et **runmqckm** (iKeycmd) ne prennent pas en charge PKCS #7 (`.p7b`) fichiers. Utilisez l'outil **runmqckm** pour importer des certificats à partir d'un fichier PKCS #7 sous UNIX, Linux, and Windows.

Utilisez la commande suivante pour ajouter un certificat d'autorité de certification à partir d'un fichier PKCS #7 :

```
runmqckm -cert -add -db filename -pw password -type cms -file filename
-label label
```

`-db` *filename*	est le nom de fichier qualifié complet de la base de données de clés CMS.
`-pw` *password*	est le mot de passe de la base de données de clés.
`-type` *cms*	est le type de la base de données de clés.
`-file` *filename*	est le nom du fichier PKCS #7 .
`-label` *label*	est le libellé auquel le certificat est affecté dans la base de données cible. Le premier certificat prend le label donné. Tous les autres certificats, s'ils sont présents, sont libellés avec leur nom de sujet.

Utilisez la commande suivante pour importer un certificat personnel à partir d'un fichier PKCS #7 :

```
runmqckm -cert -import -db filename -pw password -type pkcs7 -target filename
-target_pw password -target_type cms -label label -new_label label
```

`-db filename`	est le nom de fichier qualifié complet du fichier contenant le certificat PKCS #7 .
`-pw password`	est le mot de passe du certificat PKCS #7 .
`-type pkcs7`	est le type du fichier.
`-target filename`	est le nom de la base de données de clés de destination.
`-target_pw password`	est le mot de passe de la base de données de clés de destination.
`-target_type cms`	est le type de la base de données spécifiée par `-target`
`-label label`	est le libellé du certificat à importer.
`-new_label label`	est le libellé auquel le certificat sera affecté dans la base de données cible. Si vous omettez l'option `-new_label` , la valeur par défaut est d'utiliser la même valeur que l'option `-label` .

ULW *Suppression d'un certificat d'un référentiel de clés sur UNIX, Linux, and Windows*

Utilisez cette procédure pour supprimer des certificats personnels ou de l'autorité de certification.

Utilisation `strmqikm`

Si vous devez gérer les certificats TLS d'une manière conforme à la norme FIPS, utilisez la commande **runmqakm** . **strmqikm** (iKeyman) ne fournit pas d'option compatible FIPS.

1. Démarrez l'interface graphique à l'aide de la commande **strmqikm** (sous UNIX, Linux, and Windows).
2. Dans le menu **Key Database File**, cliquez sur **Ouvrir**. La fenêtre Ouvrir s'affiche.
3. Cliquez sur **Key database type** et sélectionnez **CMS** (système de gestion des certificats).
4. Cliquez sur **Parcourir** pour accéder au répertoire contenant le fichier de clés.
5. Sélectionnez le fichier de base de données de clés à partir duquel vous souhaitez supprimer le certificat, par exemple key **.** kdb.
6. Cliquez sur **Ouvrir**. La fenêtre Password Prompt s'ouvre.
7. Entrez le mot de passe défini lors de la création de la base de données de clés et cliquez sur **OK**. Le nom de votre fichier de base de données de clés s'affiche dans la zone **Nom de fichier** .
8. Dans la liste déroulante, sélectionnez **Certificats personnels** ou **Certificats de signataire** .
9. Sélectionnez le certificat à supprimer.
10. Si vous ne disposez pas encore d'une copie du certificat et que vous souhaitez l'enregistrer, cliquez sur **Exporter / Importer** et exportez-la (voir «Exportation d'un certificat personnel à partir d'un référentiel de clés sous UNIX, Linux, and Windows», à la page 315).
11. Une fois le certificat sélectionné, cliquez sur **Supprimer**. La fenêtre de confirmation s'ouvre.
12. Cliquez sur **Oui**. La zone **Certificats personnels** n'affiche plus le libellé du certificat que vous avez supprimé.

Utilisation de la ligne de commande

Utilisez les commandes suivantes pour supprimer un certificat à l'aide de **runmqckm**:

- Sous UNIX, Linux, and Windows :

```
runmqckm -cert -delete -db filename -pw password -label label
```

où :

-db *filename*	est le nom de fichier qualifié complet d'une base de données de clés CMS.
-pw *password*	correspond au mot de passe de la base de données de clés CMS.
-label *label*	est le label attaché au certificat personnel.
-fips	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

ULW *Génération de mots de passe fiables pour la protection des référentiels de clés sous UNIX, Linux, and Windows*

Vous pouvez générer des mots de passe fiables pour la protection du référentiel de clés à l'aide de la commande **runmqakm** (GSKCapiCmd).

Vous pouvez utiliser la commande **runmqakm** avec les paramètres suivants pour générer un mot de passe fiable:

```
runmqakm -random -create -length 14 -strong -fips
```

Lorsque vous utilisez le mot de passe généré dans le paramètre **-pw** des commandes d'administration de certificat suivantes, placez toujours le mot de passe entre guillemets. Sur les systèmes UNIX and Linux , vous devez également utiliser une barre oblique inversée pour échapper les caractères suivants s'ils apparaissent dans la chaîne de mot de passe:

```
! \ " '
```

Lorsque vous entrez le mot de passe en réponse à une invite de **runmqckm**, **runmqakm** ou de l'interface graphique **strmqikm** , il n'est pas nécessaire de le citer ou de le mettre en échappement. Elle n'est pas nécessaire car l'interpréteur de commandes du système d'exploitation n'affecte pas la saisie des données dans ces cas.

ULW *Configuration du matériel de cryptographie sous UNIX, Linux, and Windows*

Vous pouvez configurer le matériel de cryptographie pour un gestionnaire de files d'attente ou un client de plusieurs manières.

Vous pouvez configurer le matériel de cryptographie pour un gestionnaire de files d'attente sur UNIX, Linux, and Windows à l'aide de l'une des méthodes suivantes:

- Utilisez la commande ALTER QMGR MQSC avec le paramètre SSLCRYP, comme décrit dans ALTER QMGR.
- Utilisez IBM MQ Explorer pour configurer le matériel de cryptographie sur votre système UNIX, Linux ou Windows . Pour plus d'informations, reportez-vous à l'aide en ligne.

Vous pouvez configurer le matériel de cryptographie pour un client IBM MQ sous UNIX, Linux, and Windows à l'aide de l'une des méthodes suivantes:

- Définissez la variable d'environnement MQSSLCRYP. Les valeurs autorisées pour MQSSLCRYP sont les mêmes que pour le paramètre SSLCRYP, comme décrit dans ALTER QMGR.

  Si vous utilisez la version GSK_PKCS11 du paramètre SSLCRYP, le libellé de jeton PKCS #11 doit correspondre au libellé avec lequel vous avez configuré votre matériel.
- Définissez la zone **CryptoHardware** de la structure des options de configuration SSL, MQSCO, sur un appel MQCONNX. Pour plus d'informations, voir Présentation de MQSCO.

Si vous avez configuré du matériel cryptographique qui utilise l'interface PKCS #11 à l'aide de l'une de ces méthodes, vous devez stocker le certificat personnel à utiliser sur vos canaux dans le fichier de la base de données de clés pour le jeton cryptographique que vous avez configuré. Ceci est décrit dans «Gestion des certificats sur le matériel PKCS #11», à la page 323.

ULW *Gestion des certificats sur le matériel PKCS #11*

Vous pouvez gérer des certificats numériques sur du matériel de cryptographie qui prend en charge l'interface PKCS #11 .

Pourquoi et quand exécuter cette tâche

Vous devez créer une base de données de clés pour préparer l'environnement IBM MQ , même si vous n'avez pas l'intention d'y stocker des certificats d'autorité de certification, mais que vous stockez tous vos certificats sur votre matériel de cryptographie. Une base de données de clés est nécessaire pour que le gestionnaire de files d'attente y fasse référence dans sa zone SSLKEYR ou pour que l'application client y fasse référence dans la variable d'environnement MQSSLKEYR. Cette base de données de clés est également requise si vous créez une demande de certificat.

Vous créez la base de données de clés à l'aide de la ligne de commande ou de l'interface utilisateur **strmqikm** (iKeyman).

Procédure

Créez une base de données de clés à l'aide de la ligne de commande.

1. Exécutez l'une des commandes suivantes:
   - A l'aide de **runmqckm**:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - A l'aide de **runmqakm**:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
     -stash -fips -strong
     ```

   où :

   -db *nom_fichier*
   Indique le nom de fichier qualifié complet d'une base de données de clés CMS et doit avoir l'extension de fichier .kdb.

   -pw *mot_de_passe*
   Indique le mot de passe de la base de données de clés CMS.

   -type *cms*
   Indique le type de base de données. (Pour IBM MQ, il doit s'agir de cms.)

   -stash
   Sauvegarde le mot de passe de la base de données de clés dans un fichier.

   -fips
   indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

   -forte
   Vérifie que le mot de passe entré répond aux exigences minimales de puissance de mot de passe. Les exigences minimales pour un mot de passe sont les suivantes:

   - Le mot de passe doit avoir une longueur minimale de 14 caractères.
   - Le mot de passe doit contenir au moins un caractère minuscule, un caractère majuscule et un chiffre ou un caractère spécial. Les caractères spéciaux incluent l'astérisque (*), le signe dollar

($), le signe nombre (#) et le signe pourcentage (%). Un espace est classé comme un caractère spécial.

- Chaque caractère peut apparaître au maximum trois fois dans un mot de passe.
- Un maximum de deux caractères consécutifs dans le mot de passe peut être identique.
- Tous les caractères se trouvent dans le jeu de caractères ASCII imprimables standard, compris entre 0x20 et 0x7E.

Vous pouvez également créer une base de données de clés à l'aide de l'interface utilisateur **strmqikm** (iKeyman).

2. Sur les systèmes UNIX and Linux , connectez-vous en tant que superutilisateur. Sur les systèmes Windows , connectez-vous en tant qu'administrateur ou en tant que membre du groupe MQM.
3. Ouvrez le fichier de propriétés de sécurité Java, `java.security`.

   - Sur les systèmes UNIX and Linux , le fichier de propriétés de sécurité Java se trouve dans le sous-répertoire `java/jre64/jre/lib/security` du répertoire d'installation IBM MQ .
   - Sur les systèmes Windows , le fichier de propriétés de sécurité Java se trouve dans le sous-répertoire `java\jre\lib\security` du répertoire d'installation IBM MQ .

   S'il n'est pas déjà présent dans le fichier, ajoutez le fournisseur de sécurité IBMPKCS11Impl . Par exemple, en ajoutant la ligne suivante:

   ```
   security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl
   ```

4. Démarrez l'interface utilisateur en exécutant la commande **strmqikm** .
5. Cliquez sur **Fichier de base de données de clés** > **Ouvrir**.
6. Cliquez sur **Type de base de données de clés** et sélectionnez **PKCS11Direct**.
7. Dans la zone **Nom de fichier** , entrez le nom du module de gestion de votre matériel de cryptographie ; par exemple, `PKCS11_API.so`.

   Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.
8. Dans la zone **Emplacement** , entrez le chemin:

   - Sur les systèmes UNIX and Linux , il peut s'agir de `/usr/lib/pksc11`, par exemple.
   - Sur les systèmes Windows , vous pouvez entrer le nom de la bibliothèque ; par exemple, `cryptoki`.

   Cliquez sur **OK**. La fenêtre Open Cryptographic Token s'ouvre.
9. Sélectionnez le libellé de jeton de périphérique cryptographique que vous souhaitez utiliser pour stocker les certificats.
10. Dans la zone **Cryptographic Token Password** , entrez le mot de passe que vous avez défini lors de la configuration du matériel cryptographique.
11. Si votre matériel de cryptographie a la capacité de contenir les certificats de signataire requis pour recevoir ou importer un certificat personnel, décochez les deux cases de la base de données de clés secondaire et passez à l'étape «15», à la page 325.

    Si vous avez besoin d'une base de données de clés CMS secondaire pour stocker les certificats de signataire, sélectionnez **Ouvrir un fichier de base de données de clés secondaire existant** ou **Créer un fichier de base de données de clés secondaire**.
12. Dans la zone **Nom de fichier** , entrez un nom de fichier. Cette zone contient déjà le texte `key.kdb`. Si votre nom de radical est `key`, laissez cette zone inchangée. Si vous avez spécifié un autre nom de radical, remplacez `key` par votre nom de radical. Vous ne devez pas modifier le suffixe `.kdb` .
13. Dans la zone **Emplacement** , entrez le chemin, par exemple:

- Pour un gestionnaire de files d'attente: `/var/mqm/qmgrs/QM1/ssl`
- Pour un IBM MQ MQI client: `/var/mqm/ssl`

Cliquez sur **OK**. La fenêtre Password Prompt s'ouvre.

14. Entrez un mot de passe.

    Si vous avez sélectionné **Ouvrir un fichier de base de données de clés secondaire existant** à l'étape «11», à la page 324, entrez un mot de passe dans la zone **Mot de passe** .

    Si vous avez sélectionné **Créer un fichier de base de données de clés secondaires** à l'étape «11», à la page 324, effectuez les sous-étapes suivantes:

    a) Entrez un mot de passe dans la zone **Mot de passe** , puis entrez-le à nouveau dans la zone **Confirmer le mot de passe** .

    b) Sélectionnez **Stocker le mot de passe dans un fichier**. Notez que si vous ne stockez pas le mot de passe, les tentatives de démarrage des canaux TLS échouent car ils ne peuvent pas obtenir le mot de passe requis pour accéder au fichier de la base de données de clés.

    c) Cliquez sur **OK**. Une fenêtre s'ouvre, confirmant que le mot de passe se trouve dans le fichier `key.sth` (sauf si vous avez spécifié un nom de radical différent).

15. Cliquez sur **OK**. Le cadre de contenu de la base de données de clés s'affiche.

ULW *Demande d'un certificat personnel pour votre matériel PKCS #11*

Utilisez cette procédure pour un gestionnaire de files d'attente ou un IBM MQ MQI client afin de demander un certificat personnel pour votre matériel de cryptographie.

Pourquoi et quand exécuter cette tâche

Cette tâche explique comment utiliser l'interface utilisateur **strmqikm** pour demander un certificat personnel. Si vous utilisez l'interface de ligne de commande, voir «Utilisation de la ligne de commande», à la page 307.

Remarque : IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .

Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.

Procédure

Pour demander un certificat personnel à partir de l'interface utilisateur **strmqikm** (iKeyman), procédez comme suit:

1. Procédez comme suit pour utiliser votre matériel de cryptographie. Voir «Gestion des certificats sur le matériel PKCS #11», à la page 323.
2. Dans le menu **Créer** , cliquez sur **Nouvelle demande de certificat**.

   La fenêtre Créer une nouvelle clé et une nouvelle demande de certificat s'ouvre.
3. Dans la zone **Key Label** , entrez le libellé du certificat.

   Le libellé est soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmwebspheremq` avec le nom du gestionnaire de files d'attente ou l'ID utilisateur de connexion IBM MQ MQI client ajouté, le tout en minuscules. Pour plus de détails voir Labels de certificat numérique.
4. Sélectionnez la **Taille de clé** et l' **algorithme de signature** dont vous avez besoin.
5. Entrez des valeurs pour **Nom usuel** et **Organisation**, puis sélectionnez un **pays**. Pour les autres zones facultatives, acceptez les valeurs par défaut ou entrez ou sélectionnez de nouvelles valeurs.

   Notez que vous ne pouvez indiquer qu'un seul nom dans la zone **Unité organisationnelle** . Pour plus d'informations sur ces zones, voir «Noms distinctifs», à la page 11.
6. Dans la zone **Entrez le nom d'un fichier dans lequel stocker la demande de certificat** , acceptez la valeur par défaut `certreq.arm`ou entrez une nouvelle valeur avec un chemin d'accès complet.

7. Cliquez sur **OK**.

   Une fenêtre de confirmation s'ouvre.
8. Cliquez sur **OK**.

   La liste **Demandes de certificat personnel** affiche le libellé de la nouvelle demande de certificat personnel que vous avez créée. La demande de certificat est stockée dans le fichier que vous avez choisi à l'étape «6», à la page 325.
9. Demandez le nouveau certificat personnel soit en envoyant le fichier à une autorité de certification, soit en copiant le fichier dans le formulaire de demande sur le site Web de l'autorité de certification.

ULW *Réception d'un certificat personnel dans votre matériel PKCS #11*

Utilisez cette procédure pour un gestionnaire de files d'attente ou un IBM MQ MQI client afin de recevoir un certificat personnel sur votre matériel de cryptographie.

Avant de commencer

Ajoutez le certificat de l'autorité de certification qui a signé le certificat personnel. Ajoutez-le dans le matériel de cryptographie ou dans la base de données de clés CMS secondaire. Effectuez cette opération avant de recevoir le certificat signé dans le matériel de cryptographie. Pour ajouter un certificat d'autorité de certification à un fichier de clés, suivez la procédure décrite dans «Ajout d'un certificat de l'autorité de certification ou de la partie publique d'un certificat autosigné dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 314.

Procédure

- Pour recevoir un certificat personnel à l'aide de l'interface utilisateur **strmqikm** (iKeyman), procédez comme suit:
  a) Procédez comme suit pour utiliser votre matériel de cryptographie. Voir «Gestion des certificats sur le matériel PKCS #11», à la page 323.
  b) Cliquez sur **Recevoir**. La fenêtre Recevoir un certificat d'un fichier s'affiche.
  c) Entrez le nom du fichier de certificat et l'emplacement du nouveau certificat personnel ou cliquez sur **Parcourir** pour sélectionner le nom et l'emplacement.
  d) Cliquez sur **OK**. Si vous disposez déjà d'un certificat personnel dans votre base de données de clés, une fenêtre s'ouvre et vous demande si vous souhaitez définir la clé que vous ajoutez comme clé par défaut dans la base de données.
  e) Cliquez sur **Oui** ou sur **Non**. La fenêtre Enter a Label s'ouvre.
  f) Cliquez sur **OK**. La liste **Certificats personnels** affiche le libellé du nouveau certificat personnel que vous avez ajouté. Ce libellé est formé en ajoutant le libellé de jeton de chiffrement avant le libellé que vous avez fourni.
- Pour recevoir un certificat personnel à l'aide de la commande **runmqakm** (GSKCapiCmd), procédez comme suit:
  a) Ouvrez une fenêtre de commande configurée pour votre environnement.
  b) Recevez le certificat personnel à l'aide de la commande **runmqakm** (GSKCapiCmd):

```
runmqakm -cert -receive -file filename -crypto module_name
         -tokenlabel hardware_token -pw hardware_password
         -format cert_format -fips
         -secondaryDB filename -secondaryDBpw password
```

  où :

  -file *nom fichier*
  Indique le nom de fichier complet du fichier contenant le certificat personnel.

  -crypto *nom_module*
  Indique le nom qualifié complet de la bibliothèque PKCS #11 fournie avec le matériel de cryptographie.

-tokenlabel ***jeton_matériel***
Indique le libellé du jeton de l'unité de chiffrement PKCS #11 .

-pw ***mot_de_passe_matériel***
Indique le mot de passe pour l'accès au matériel de cryptographie.

-format ***format_certificat***
Indique le format du certificat. La valeur peut-être `ascii` pour les données ASCII codées en base 64 ou `binary` pour les données Binary DER. La valeur par défaut est ASCII.

-fips
indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.

-secondaryDB ***nom_fichier***
Indique le nom de fichier qualifié complet de la base de données de clés CMS.

-secondaryDBpw ***mot_de_passe***
Indique le mot de passe de la base de données de clés CMS.

MQ Appliance Utilisation de SSL/TLS sous IBM MQ Appliance

IBM MQ Appliance prend en charge le protocole TLS (Transport Layer Security).

Le IBM MQ Appliance comporte des commandes distinctes pour la gestion des certificats. Pour des informations détaillées sur la gestion des certificats, voir la documentation IBM MQ Appliance , Gestion des certificats TLS

z/OS Utilisation de SSL/TLS sous z/OS

Ces informations décrivent comment configurer et utiliser le protocole TLS (Transport Layer Security) sur z/OS.

Chaque rubrique contient des exemples d'exécution de chaque tâche à l'aide de RACF. Vous pouvez effectuer des tâches similaires à l'aide des autres gestionnaires de sécurité externes.

Sous z/OS, vous devez également définir le nombre de sous-tâches de serveur utilisées par chaque gestionnaire de files d'attente pour le traitement des appels TLS, comme décrit dans «Définition du paramètre SSLTASKS sous z/OS», à la page 328.

z/OS La prise en charge de TLS fait partie intégrante du système d'exploitation et est appelée *System SSL*. System SSL fait partie de l'élément Cryptographic Services Base de z/OS. Les membres Cryptographic Services Base sont installés dans le fichier *pdsname*. Fichier partitionné (PDS) SIEALNKE. Lorsque vous installez System SSL, veillez à choisir les options appropriées pour fournir les CipherSpecs dont vous avez besoin.

z/OS *Exigences d'ID utilisateur supplémentaires pour TLS sur z/OS*

Ces informations décrivent les exigences supplémentaires dont votre ID utilisateur a besoin pour configurer et utiliser TLS sur z/OS.

Vérifiez que vous disposez de toutes les mises à jour HIPER (High Impact or ???) appropriées sur votre système.

Vérifiez que vous avez configuré les prérequis suivants:

- L'ID utilisateur *ssidCHIN* est défini correctement dans RACFet l'ID utilisateur *ssidCHIN* dispose d'un accès en lecture (READ) aux profils suivants:
  - IRR.DIGTCERT.LIST
  - IRR.DIGTCERT.LISTRING

  Ces variables sont définies dans la classe RACF FACILITY.
- L'ID utilisateur *ssidCHIN* est le propriétaire du fichier de clés.

- Le certificat personnel du gestionnaire de files d'attente, s'il est créé par la commande RACDCERT, est créé avec un ID utilisateur de type de certificat qui est également identique à l'ID utilisateur *ssidCHIN* .
- L'initiateur de canal est recyclé ou la commande **REFRESH SECURITY TYPE(SSL)** est émise pour récupérer les modifications apportées au fichier de clés.
- La procédure de l'initialisateur de canal IBM MQ a accès à la bibliothèque d'exécution SSL système *pdsname*.SIEALNKE via la liste de liens, LPA ou une instruction de définition de données STEPLIB. Cette bibliothèque doit disposer de droits APF.
- L'ID utilisateur sous lequel l'initiateur de canal est exécuté est configuré pour utiliser UNIX System Services (USS), comme décrit dans la documentation z/OS UNIX System Services Planning.

  Les utilisateurs qui ne souhaitent pas que l'initiateur de canal appelle UNIX System Services à l'aide de l'ID utilisateur par défaut et du segment OMVS doivent uniquement modéliser un nouveau segment OMVS basé sur le segment par défaut car l'initiateur de canal ne requiert pas de droits spéciaux et ne s'exécute pas dans UNIX en tant que superutilisateur.

z/OS *Définition du paramètre SSLTASKS sous z/OS*

Utilisez la commande ALTER QMGR pour définir le nombre de sous-tâches du serveur pour le traitement des appels TLS

Pour utiliser les canaux TLS, assurez-vous qu'il existe au moins deux sous-tâches de serveur en définissant le paramètre SSLTASKS à l'aide de la commande ALTER QMGR. Exemple :

```
ALTER QMGR SSLTASKS(5)
```

Pour éviter les problèmes d'allocation de mémoire, ne définissez pas l'attribut SSLTASKS sur une valeur supérieure à huit dans un environnement où il n'y a pas de vérification de la liste de révocation de certificat (CRL).

Si la vérification CRL est utilisée, une SSLTASK est conservée par le canal concerné pendant la durée de cette vérification. Il peut s'agir d'un temps écoulé significatif lors du contact du serveur LDAP approprié, car chaque SSLTASK est un bloc de contrôle de tâche z/OS .

Vous devez redémarrer l'initiateur de canal si vous modifiez la valeur de l'attribut SSLTASKS.

z/OS *Configuration d'un référentiel de clés sur z/OS*

Configurez un référentiel de clés aux deux extrémités de la connexion. Associez chaque référentiel de clés à son gestionnaire de files d'attente.

Une connexion TLS requiert un *référentiel de clés* à chaque extrémité de la connexion. Chaque gestionnaire de files d'attente doit avoir accès à un référentiel de clés. Utilisez le paramètre SSLKEYR dans la commande ALTER QMGR pour associer un référentiel de clés à un gestionnaire de files d'attente. Pour plus d'informations, voir «Référentiel de clés SSL/TLS», à la page 25.

Sous z/OS, les certificats numériques sont stockés dans un *fichier de clés* géré par le gestionnaire de sécurité externe (ESM). Ces certificats numériques comportent des libellés qui associent le certificat à un gestionnaire de files d'attente. TLS utilise ces certificats à des fins d'authentification. Tous les exemples qui suivent utilisent les commandes RACF . Des commandes équivalentes existent pour d'autres programmes ESM.

Sous z/OS, IBM MQ utilise soit la valeur de l'attribut **CERTLABL** , s'il est défini, soit la valeur par défaut `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ajouté. Pour plus de détails voir Labels de certificat numérique.

Le nom du référentiel de clés d'un gestionnaire de files d'attente est le nom d'un fichier de clés dans votre base de données RACF . Vous pouvez spécifier le nom du fichier de clés avant ou après sa création.

Utilisez la procédure suivante pour créer un fichier de clés pour un gestionnaire de files d'attente:

1. Vérifiez que vous disposez des droits appropriés pour émettre la commande RACDCERT (voir *SecureWay Security Server RACF Command Language Reference* pour plus de détails).

2. Entrez la commande suivante :

```
RACDCERT ID( userid1 ) ADDRING( ring-name )
```

où :

- *userid1* est l'ID utilisateur de l'espace adresse de l'initiateur de canal ou l'ID utilisateur qui va posséder le fichier de clés (si le fichier de clés est partagé).
- *ring-name* est le nom que vous souhaitez attribuer à votre fichier de clés. La longueur de ce nom peut atteindre 237 caractères. Ce nom est sensible à la casse. Spécifiez *ring-name* en majuscules pour éviter les problèmes.

z/OS *Mise à disposition des certificats de l'autorité de certification pour un gestionnaire de files d'attente sous z/OS*

Une fois que vous avez créé votre fichier de clés, connectez les certificats de l'autorité de certification appropriés.

Si vous disposez du certificat de l'autorité de certification dans un fichier, vous devez d'abord ajouter le certificat à la base de données RACF à l'aide de la commande suivante:

```
RACDCERT ID( userid1 ) ADD( input-data-set-name ) WITHLABEL( 'My CA' )
```

Ensuite, pour connecter un certificat d'autorité de certification pour My CA à votre fichier de clés, utilisez la commande suivante:

```
RACDCERT ID(userid1)
CONNECT(CERTAUTH LABEL('My CA') RING(ring-name) USAGE(CERTAUTH))
```

où *userid1* est l'ID utilisateur de l'initiateur de canal ou le propriétaire d'un fichier de clés partagé.

Pour plus d'informations sur les certificats de l'autorité de certification, voir «Certificats numériques», à la page 9.

z/OS *Localisation du référentiel de clés d'un gestionnaire de files d'attente sous z/OS*

Utilisez cette procédure pour obtenir l'emplacement du fichier de clés de votre gestionnaire de files d'attente.

1. Affichez les attributs de votre gestionnaire de files d'attente à l'aide de l'une des commandes MQSC suivantes:

```
DISPLAY QMGR ALL
DISPLAY QMGR SSLKEYR
```

2. Examinez le résultat de la commande pour déterminer l'emplacement du fichier de clés.

z/OS *Spécification de l'emplacement du référentiel de clés pour un gestionnaire de files d'attente sous z/OS*

Pour spécifier l'emplacement du fichier de clés de votre gestionnaire de files d'attente, utilisez la commande ALTER QMGR MQSC pour définir l'attribut de référentiel de clés de votre gestionnaire de files d'attente.

Exemple :

```
ALTER QMGR SSLKEYR(CSQ1RING)
```

si le fichier de clés est détenu par l'espace adresse de l'initiateur de canal, ou:

```
ALTER QMGR SSLKEYR(userid1/CSQ1RING)
```

s'il s'agit d'un fichier de clés partagé, où *userid1* est l'ID utilisateur propriétaire du fichier de clés.

z/OS *Attribution à l'initiateur de canal des droits d'accès corrects sur z/OS*

L'initiateur de canal (CHINIT) doit accéder au référentiel de clés et à certains profils de sécurité.

Octroi à CHINIT de l'accès en lecture au référentiel de clés

Si le référentiel de clés appartient à l'ID utilisateur CHINIT, cet ID utilisateur doit disposer d'un accès en lecture à l'IRR IRR.DIGTCERT.LISTRING dans la classe FACILITY et mettez à jour les droits d'accès dans le cas contraire. Accordez l'accès à l'aide de la commande PERMIT avec ACCESS (UPDATE) ou ACCESS (READ), selon le cas:

```
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID( userid ) ACCESS(UPDATE)
```

où *userid* est l'ID utilisateur de l'espace adresse de l'initiateur de canal.

Octroi de l'accès en lecture CHINIT aux profils CSF* appropriés

Pour le support matériel fourni via la fonction ICSF (Integrated Cryptographic Service Facility) à utiliser, vérifiez que votre ID utilisateur CHINIT dispose d'un accès en lecture aux profils CSF* appropriés dans la classe CSFSERV à l'aide de la commande suivante:

```
PERMIT csf-resource CLASS(CSFSERV) ID( userid ) ACCESS(READ)
```

où *csf-resource* est le nom du profil CSF* et *userid* est l'ID utilisateur de l'espace adresse de l'initiateur de canal.

Répétez cette commande pour chacun des profils CSF* suivants:

- CSFDSG
- CSFDSV
- CSFPKD
- CSFPKE
- CSFPKI

Votre ID utilisateur CHINIT peut également avoir besoin d'un accès en lecture à d'autres profils CSF*. Par exemple, si vous utilisez la spécification de chiffrement ECDHE_RSA_AES_256_GCM_SHA384 , votre ID utilisateur CHINIT doit également disposer d'un accès en lecture aux profils CSF* suivants:

- CSF1DVK
- CSF1GAV
- CSF1GKP
- CSF1SKE
- CSF1TRC
- CSF1TRD

Pour plus d'informations, voir RACF CSFSERV resource requirements.

Si vos clés de certificat sont stockées dans ICSF et que votre installation a établi un contrôle d'accès sur les clés stockées dans ICSF, vérifiez que votre ID utilisateur CHINIT dispose d'un accès en lecture au profil dans la classe CSFKEYS à l'aide de la commande suivante:

```
PERMIT IRR.DIGTCERT. userid.* CLASS(CSFKEYS) ID( userid ) ACCESS(READ)
```

où *userid* est l'ID utilisateur de l'espace adresse de l'initiateur de canal.

Utilisation de la fonction ICSF (Integrated Cryptographic Service Facility)

L'initiateur de canal peut utiliser ICSF pour générer un nombre aléatoire lors de la distribution de l'algorithme de protection par mot de passe afin de brouiller les mots de passe transitant sur les canaux client si TLS n'est pas utilisé.

Pour plus d'informations, voir «Utilisation de la fonction ICSF (Integrated Cryptographic Service Facility)», à la page 271

z/OS *Lorsque les modifications apportées aux certificats ou au référentiel de clés prennent effet sur z/OS*

Les modifications prennent effet lorsque l'initiateur de canal démarre ou que le référentiel est actualisé.

En particulier, les modifications apportées aux certificats dans le fichier de clés et à l'attribut de référentiel de clés prennent effet à l'une des occasions suivantes:

- Lorsque l'initiateur de canal est démarré ou redémarré.
- Lorsque la commande REFRESH SECURITY TYPE (SSL) est émise pour actualiser le contenu du référentiel de clés.

z/OS *Création d'un certificat personnel autosigné sur z/OS*

Utilisez cette procédure pour créer un certificat personnel autosigné.

1. Générez un certificat et une paire de clés publique et privée à l'aide de la commande suivante:

```
RACDCERT ID(userid2) GENCERT
SUBJECTSDN(CN('common-name')
           T('title')
           OU('organizational-unit')
           O('organization')
           L('locality')
           SP('state-or-province')
           C('country'))
WITHLABEL('label-name')
```

2. Connectez le certificat à votre fichier de clés à l'aide de la commande suivante:

```
RACDCERT ID(userid1)
CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
```

où :

- *userid1* est l'ID utilisateur de l'espace adresse de l'initiateur de canal ou du propriétaire du fichier de clés partagé.
- *userid2* est l'ID utilisateur associé au certificat et doit correspondre à l'ID utilisateur de l'espace adresse de l'initiateur de canal.

  userid1 et *userid2* peuvent avoir le même ID.
- *ring-name* est le nom que vous avez donné au fichier de clés dans «Configuration d'un référentiel de clés sur z/OS», à la page 328.
- *label-name* doit correspondre à la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, ou à la valeur par défaut `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ajouté. Pour plus de détails voir Labels de certificat numérique.

z/OS *Demande d'un certificat personnel sur z/OS*

Demandez un certificat personnel à l'aide de RACF.

Pour demander un certificat personnel, utilisez RACF comme suit:

1. Créez un certificat personnel autosigné, comme dans «Création d'un certificat personnel autosigné sur z/OS», à la page 331. Ce certificat fournit à la demande les valeurs d'attribut du nom distinctif.
2. Créez une demande de certificat PKCS #10 Base64-encoded écrite dans un fichier, à l'aide de la commande suivante:

   ```
   RACDCERT ID(userid2) GENREQ(LABEL(' label_name ')) DSN(' output_data_set_name ')
   ```

   Où

   - *userid2* est l'ID utilisateur associé au certificat et doit correspondre à l'ID utilisateur de l'espace adresse de l'initiateur de canal.
   - *label_name* est le libellé utilisé lors de la création du certificat autosigné

   Voir «Labels de certificat numérique, compréhension des exigences», à la page 26 pour des détails.
3. Envoyez le fichier à une autorité de certification pour demander un nouveau certificat personnel.
4. Lorsque le certificat signé vous est renvoyé par l'autorité de certification, ajoutez-le à nouveau dans la base de données RACF à l'aide du libellé d'origine, comme décrit dans «Ajout de certificats personnels à un référentiel de clés sous z/OS», à la page 333.

z/OS *Création d'un certificat personnel signé RACF*

RACF peut fonctionner en tant qu'autorité de certification et émettre son propre certificat d'autorité de certification.

Cette section utilise le terme *certificat de signataire* pour désigner un certificat de l'autorité de certification émis par RACF.

La clé privée du certificat de signataire doit se trouver dans la base de données RACF avant d'exécuter la procédure suivante:

1. Utilisez la commande suivante pour générer un certificat personnel signé par RACF, à l'aide du certificat de signataire contenu dans votre base de données RACF :

   ```
   RACDCERT ID(userid2) GENCERT
   SUBJECTSDN(CN('common-name')
              T('title')
              OU('organizational-unit')
              O('organization')
              L('locality')
              SP('state-or-province')
              C('country'))
   WITHLABEL('label-name')
   SIGNWITH(CERTAUTH LABEL('signer-label'))
   ```

2. Connectez le certificat à votre fichier de clés à l'aide de la commande suivante:

   ```
   RACDCERT ID(userid1)
   CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
   ```

où :

- *userid1* est l'ID utilisateur de l'espace adresse de l'initiateur de canal ou du propriétaire du fichier de clés partagé.
- *userid2* est l'ID utilisateur associé au certificat et doit correspondre à l'ID utilisateur de l'espace adresse de l'initiateur de canal.

  userid1 et *userid2* peuvent avoir le même ID.
- *ring-name* est le nom que vous avez donné au fichier de clés dans «Configuration d'un référentiel de clés sur z/OS», à la page 328.

- *label-name* doit être la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, ou la valeur par défaut `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente ajouté. Pour plus de détails voir Labels de certificat numérique.
- *signer-label* est le libellé de votre propre certificat de signataire.

z/OS *Ajout de certificats personnels à un référentiel de clés sous z/OS*

Utilisez cette procédure pour ajouter ou importer un certificat personnel dans un fichier de clés.

Une fois que l'autorité de certification vous a envoyé un nouveau certificat personnel, ajoutez-le au fichier de clés à l'aide de la procédure suivante:

1. Ajoutez le certificat à la base de données RACF à l'aide de la commande suivante:

```
RACDCERT ID( userid2 ) ADD( input-data-set-name ) WITHLABEL(' label-name ')
```

2. Connectez le certificat à votre fichier de clés à l'aide de la commande suivante:

```
RACDCERT ID( userid1 )
CONNECT(ID( userid2 ) LABEL(' label-name ') RING( ring-name ) USAGE(PERSONAL))
```

où :

- *userid1* est l'ID utilisateur de l'espace adresse de l'initiateur de canal ou du propriétaire du fichier de clés partagé.
- *userid2* est l'ID utilisateur associé au certificat et doit correspondre à l'ID utilisateur de l'espace adresse de l'initiateur de canal.
- *ring-name* est le nom que vous avez donné au fichier de clés dans «Configuration d'un référentiel de clés sur z/OS», à la page 328.
- *input-data-set-name* est le nom du fichier contenant le certificat signé par l'autorité de certification. Le fichier doit être catalogué et ne doit pas être un fichier partitionné ou un membre d'un fichier partitionné. Le format d'enregistrement (RECFM) attendu par RACDCERT est VB. RACDCERT alloue et ouvre dynamiquement le fichier et lit le certificat qu'il contient en tant que données binaires.
- *label-name* est le nom de libellé utilisé lors de la création de la demande d'origine. Il doit s'agir de la valeur de l'attribut IBM MQ **CERTLABL** , s'il est défini, ou de la valeur par défaut `ibmWebSphereMQ` avec le nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente ajouté. Pour plus de détails voir Labels de certificat numérique.

z/OS *Exportation d'un certificat personnel à partir d'un référentiel de clés sous z/OS*

Exportez le certificat à l'aide de la commande RACDCERT.

Sur le système à partir duquel vous souhaitez exporter le certificat, utilisez la commande suivante:

```
RACDCERT ID(userid2) EXPORT(LABEL('label-name'))
DSN(output-data-set-name) FORMAT(CERTB64)
```

où :

- *userid2* est l'ID utilisateur sous lequel le certificat a été ajouté au fichier de clés.
- *label-name* est le libellé du certificat que vous souhaitez extraire.
- *output-data-set-name* est le fichier dans lequel le certificat est placé.
- CERTB64 est un certificat X.509 codé DER au format Base64 . Vous pouvez choisir un autre format, par exemple:

  CERTDER
  Certificat X.509 codé DER au format binaire

PKCS12B64
Certificat PKCS #12 au format Base64

PKCS12DER
Certificat PKCS #12 au format binaire

z/OS *Suppression d'un certificat personnel d'un référentiel de clés sous z/OS*

Supprimez un certificat personnel à l'aide de la commande RACDCERT.

Avant de supprimer un certificat personnel, vous pouvez en sauvegarder une copie. Pour copier votre certificat personnel dans un fichier avant de le supprimer, suivez la procédure décrite dans «Exportation d'un certificat personnel à partir d'un référentiel de clés sous z/OS», à la page 333. Utilisez ensuite la commande suivante pour supprimer votre certificat personnel:

```
RACDCERT ID( userid2 ) DELETE(LABEL(' label-name '))
```

où :

- *userid2* est l'ID utilisateur sous lequel le certificat a été ajouté au fichier de clés.
- *label-name* est le nom du certificat à supprimer.

z/OS *Attribution d'un nouveau nom à un certificat personnel dans un référentiel de clés sous z/OS*

Renommez un certificat à l'aide de la commande RACDCERT.

Si vous ne souhaitez pas qu'un certificat avec un libellé spécifique soit trouvé, mais que vous ne souhaitez pas le supprimer, vous pouvez le renommer temporairement à l'aide de la commande suivante:

```
RACDCERT ID( userid2 ) LABEL(' label-name ') NEWLABEL(' new-label-name ')
```

où :

- *userid2* est l'ID utilisateur sous lequel le certificat a été ajouté au fichier de clés.
- *label-name* est le nom du certificat que vous souhaitez renommer.
- *new-label-name* est le nouveau nom du certificat.

Cela peut être utile lors du test de l'authentification du client TLS.

z/OS *Association d'un ID utilisateur à un certificat numérique sur z/OS*

IBM MQ peut utiliser un ID utilisateur associé à un certificat RACF en tant qu'ID utilisateur de canal. Associez un ID utilisateur à un certificat en l'installant sous cet ID utilisateur ou en utilisant un filtre de nom de certificat.

La méthode décrite dans cette rubrique est une alternative à la méthode indépendante de la plateforme pour associer un ID utilisateur à un certificat numérique, qui utilise des enregistrements d'authentification de canal. Pour plus d'informations sur les enregistrements d'authentification de canal, voir «Enregistrements d'authentification de canal», à la page 50.

Lorsqu'une entité à une extrémité d'un canal TLS reçoit un certificat d'une connexion distante, l'entité demande à RACF si un ID utilisateur est associé à ce certificat. L'entité utilise cet ID utilisateur comme ID utilisateur de canal. Si aucun ID utilisateur n'est associé au certificat, l'entité utilise l'ID utilisateur sous lequel l'initiateur de canal s'exécute.

Associez un ID utilisateur à un certificat de l'une des manières suivantes:

- Installez ce certificat dans la base de données RACF sous l'ID utilisateur auquel vous souhaitez l'associer, comme décrit dans «Ajout de certificats personnels à un référentiel de clés sous z/OS», à la page 333.

- Utilisez un filtre de nom de certificat (CNF) pour mapper le nom distinctif du sujet ou de l'émetteur du certificat à l'ID utilisateur, comme décrit dans «Configuration d'un filtre de nom de certificat sur z/OS», à la page 335.

z/OS *Configuration d'un filtre de nom de certificat sur z/OS*

Utilisez la commande RACDCERT pour définir un filtre de nom de certificat (CNF), qui mappe un nom distinctif à un ID utilisateur.

Procédez comme suit pour configurer une fonction CNF.

1. Activez les fonctions CNF à l'aide de la commande suivante. Pour ce faire, vous devez disposer du droit de mise à jour sur la classe DIGTNMAP.

```
SETROPTS CLASSACT(DIGTNMAP) RACLIST(DIGTNMAP)
```

2. Définissez le CNF. Exemple :

```
RACDCERT ID(USER1) MAP WITHLABEL('filter1') TRUST
SDNFILTER('O=IBM.C=UK') IDNFILTER('O=ExampleCA.L=Internet')
```

   où USER1 est l'ID utilisateur à utiliser lorsque:

   - Le nom distinctif du sujet a une organisation de `IBM` et un pays de UK.
   - Le nom distinctif de l'émetteur a une organisation de `ExampleCA` et une localité de `Internet`.

3. Actualisez les mappages CNF:

```
SETROPTS RACLIST(DIGTNMAP) REFRESH
```

Remarque :

1. Si le certificat réel est stocké dans la base de données RACF , l'ID utilisateur sous lequel il est installé est utilisé de préférence à l'ID utilisateur associé à un CNF. Si le certificat n'est pas stocké dans la base de données RACF , l'ID utilisateur associé au fichier CNF correspondant le plus spécifique est utilisé. Les correspondances du nom distinctif du sujet sont considérées comme plus spécifiques que les correspondances du nom distinctif de l'émetteur.
2. Les modifications apportées aux fichiers CNF ne s'appliquent pas tant que vous n'actualisez pas les mappages CNF.
3. Un nom distinctif correspond au filtre de nom distinctif dans un CNF uniquement si le filtre de nom distinctif est identique à la *partie la moins significative* du nom distinctif. La partie la moins significative du nom distinctif comprend les attributs qui sont généralement répertoriés à l'extrémité la plus à droite du nom distinctif, mais qui apparaissent au début du certificat.

   Prenons l'exemple de SDNFILTER `'O=IBM.C=UK'`. Un nom distinctif de sujet de `'CN=QM1.O=IBM.C=UK'` correspond à ce filtre, mais un nom distinctif de sujet de `'CN=QM1.O=IBM.L=Hursley.C=UK'` ne correspond pas à ce filtre.

   La partie la moins significative de certains certificats peut contenir des zones qui ne correspondent pas au filtre de nom distinctif. Envisagez d'exclure ces certificats en spécifiant un modèle de nom distinctif dans le modèle SSLPEER de la commande DEFINE CHANNEL.
4. Si le CNF correspondant le plus spécifique est défini sur RACF en tant que NOTRUST, l'entité utilise l'ID utilisateur sous lequel l'initiateur de canal s'exécute.
5. RACF utilise le caractère `'.'` comme séparateur. IBM MQ utilise une virgule ou un point-virgule.

Vous pouvez définir des CNF pour vous assurer que l'entité ne définit jamais l'ID utilisateur du canal sur la valeur par défaut, qui est l'ID utilisateur sous lequel l'initiateur de canal est exécuté. Pour chaque certificat de l'autorité de certification dans le fichier de clés associé à l'entité, définissez un fichier CNF avec un filtre IDNFILTER qui correspond exactement au nom distinctif du sujet de ce certificat de l'autorité de certification. Cela permet de s'assurer que tous les certificats que l'entité peut utiliser

correspondent à au moins l'un de ces fichiers CNF. En effet, tous ces certificats doivent soit être connectés au fichier de clés associé à l'entité, soit être émis par une autorité de certification pour laquelle un certificat est connecté au fichier de clés associé à l'entité.

Pour plus d'informations sur les commandes que vous utilisez pour manipuler des fichiers CNF, voir *SecureWay Security Server RACF Security Administrator's Guide* .

z/OS *Définition d'un canal émetteur et d'une file d'attente de transmission sur QMA sous z/OS*

Utilisez les commandes **DEFINE CHANNEL** et **DEFINE QLOCAL** pour configurer les objets requis.

Procédure

Sur QMA, émettez des commandes similaires à l'exemple suivant:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) DESCR('Sender channel using TLS from QMA to QMB')

DEFINE QLOCAL(QMB) USAGE(XMITQ)
```

Résultats

Un canal émetteur, TO.QMBet une file d'attente de transmission, QMB, sont créées.

z/OS *Définition d'un canal récepteur sur QMB sur z/OS*

Utilisez la commande **DEFINE CHANNEL** pour configurer l'objet requis.

Procédure

Sur QMB, exécutez une commande similaire à l'exemple suivant:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS to QMB')
```

Résultats

Un canal récepteur, TO.QMBest créé.

z/OS *Démarrage du canal émetteur sur QMA sous z/OS*

Si nécessaire, démarrez un programme d'écoute et actualisez la sécurité. Démarrez ensuite le canal à l'aide de la commande **START CHANNEL** .

Procédure

1. Facultatif : Si ce n'est pas déjà fait, démarrez un programme d'écoute sur QMB.

   Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur lorsque cela est nécessaire. Pour plus d'informations sur le démarrage d'un programme d'écoute, voir Démarrage d'un programme d'écoute de canal.
2. Facultatif : Si des canaux SSL/TLS ont déjà été exécutés, exécutez la commande REFRESH SECURITY TYPE(SSL).

   Cela garantit que toutes les modifications apportées au référentiel de clés sont disponibles.
3. Démarrez le canal sur QMA à l'aide de la commande START CHANNEL(TO.QMB).

Résultats

Le canal émetteur est démarré.

z/OS *Echange de certificats autosignés sur z/OS*

Echangez les certificats que vous avez extraits précédemment. Si vous utilisez FTP, utilisez le format correct.

Procédure

Transférez la partie CA du certificat QM1 vers le système QM2 et inversement, par exemple par FTP.

Si vous transférez les certificats à l'aide de FTP, vous devez le faire au format approprié.

Transférez les types de certificat suivants au format *binaire* :

- Binaire codé DER X.509
- PKCS #7 (certificats de l'autorité de certification)
- PKCS #12 (certificats personnels)

Transférez les types de certificat suivants au format ASCII:

- PEM (privacy-enhanced mail)
- Base64 codé X.509

z/OS *Définition d'un canal émetteur et d'une file d'attente de transmission sur QM1 sous z/OS*

Utilisez les commandes **DEFINE CHANNEL** et **DEFINE QLOCAL** pour configurer les objets requis.

Procédure

Sous QM1, exécutez des commandes similaires à l'exemple suivant:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) DESCR('Sender channel using TLS from QM1 to QM2')

DEFINE QLOCAL(QM2) USAGE(XMITQ)
```

Le CipherSpec doit être le même à chaque extrémité du canal.

Seul le paramètre SSLCIPH est obligatoire si vous souhaitez que votre canal utilise TLS. Pour plus d'informations sur les valeurs autorisées pour le paramètre SSLCIPH, voir «CipherSpecs et CipherSuites dans IBM MQ», à la page 40 .

Résultats

Un canal émetteur, QM1.TO.QM2et une file d'attente de transmission, QM2, sont créés.

z/OS *Définition d'un canal récepteur sur QM2 sous z/OS*

Utilisez la commande **DEFINE CHANNEL** pour configurer l'objet requis.

Procédure

Sous QM2, exécutez une commande similaire à l'exemple suivant:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS from QM1 to QM2')
```

Le canal doit avoir le même nom que le canal émetteur que vous avez défini dans «Définition d'un canal émetteur et d'une file d'attente de transmission sur QM1 sous z/OS», à la page 337et utiliser le même CipherSpec.

z/OS *Démarrage du canal émetteur sur QM1 sous z/OS*

Si nécessaire, démarrez un programme d'écoute et actualisez la sécurité. Démarrez ensuite le canal à l'aide de la commande **START CHANNEL** .

Procédure

1. Facultatif : Si ce n'est pas déjà fait, démarrez un programme d'écoute sur QM2.

   Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur lorsque cela est nécessaire. Pour plus d'informations sur le démarrage d'un programme d'écoute, voir Démarrage d'un programme d'écoute de canal
2. Facultatif : Si des canaux SSL/TLS ont été exécutés précédemment, exécutez la commande REFRESH SECURITY TYPE (SSL).

   Cela garantit que toutes les modifications apportées au référentiel de clés sont disponibles.
3. Sur QM1, démarrez le canal à l'aide de la commande `START CHANNEL(QM1.TO.QM2)`.

Résultats

Le canal émetteur est démarré.

z/OS *Actualisation de l'environnement SSL ou TLS sous z/OS*

Actualisez l'environnement TLS sur le gestionnaire de files d'attente QMA à l'aide de la commande **REFRESH SECURITY** .

Procédure

Sur QMA, entrez la commande suivante:

```
REFRESH SECURITY TYPE(SSL)
```

Cela garantit que toutes les modifications apportées au référentiel de clés sont disponibles.

z/OS *Autorisation des connexions anonymes sur un canal récepteur sous z/OS*

Utilisez la commande **ALTER CHANNEL** pour rendre l'authentification de client SSL ou TLS facultative.

Procédure

Sur QMB, entrez la commande suivante:

```
ALTER CHANNEL(TO.QMB) CHLTYPE(RCVR) SSLCAUTH(OPTIONAL)
```

z/OS *Démarrage du canal émetteur sur QM1 sous z/OS*

Si nécessaire, démarrez l'initiateur de canal, démarrez un programme d'écoute et actualisez la sécurité. Démarrez ensuite le canal à l'aide de la commande **START CHANNEL** .

Procédure

1. Facultatif : Si vous ne l'avez pas déjà fait, démarrez l'initiateur de canal.
2. Facultatif : Si ce n'est pas déjà fait, démarrez un programme d'écoute sur QM2.

   Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur lorsque cela est nécessaire. Pour plus d'informations sur le démarrage d'un programme d'écoute, voir Démarrage d'un programme d'écoute de canal
3. Facultatif : Si l'initiateur de canal était déjà en cours d'exécution ou si des canaux SSL/TLS ont déjà été exécutés, exécutez la commande REFRESH SECURITY TYPE (SSL).

   Cela garantit que toutes les modifications apportées au référentiel de clés sont disponibles.

4. Sur QM1, démarrez le canal à l'aide de la commande START CHANNEL(QM1.TO.QM2).

Résultats

Le canal émetteur est démarré.

z/OS *Démarrage du canal émetteur sur QMA sous z/OS*

Si nécessaire, démarrez l'initiateur de canal, démarrez un programme d'écoute et actualisez la sécurité. Démarrez ensuite le canal à l'aide de la commande **START CHANNEL** .

Procédure

1. Facultatif : Si ce n'est pas déjà fait, démarrez l'initiateur de canal.
2. Facultatif : Si ce n'est pas déjà fait, démarrez un programme d'écoute sur QMB.

   Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur lorsque cela est nécessaire. Pour plus d'informations sur le démarrage d'un programme d'écoute, voir Démarrage d'un programme d'écoute de canal.
3. Facultatif : Si l'initiateur de canal était déjà en cours d'exécution ou si des canaux SSL/TLS ont déjà été exécutés, exécutez la commande REFRESH SECURITY TYPE(SSL).

   Cela garantit que toutes les modifications apportées au référentiel de clés sont disponibles.
4. Démarrez le canal sur QMA à l'aide de la commande START CHANNEL(TO.QMB).

Résultats

Le canal émetteur est démarré.

z/OS *Modification de la longueur de clé de courbe elliptique sous z/OS*

Comment modifier la variable d'environnement GSK_CLIENT_ECURVE_LIST, pour définir la liste des courbes elliptiques ou des groupes pris en charge spécifiés par le client, sous la forme d'une chaîne composée d'une ou de plusieurs valeurs à 4 caractères par ordre de préférence d'utilisation.

Important : Vous devez appliquer le correctif dans l' z/OS APAR OA61783 pour permettre au système d'exploitation d'appliquer certaines courbes elliptiques lors de l'utilisation de connexions négociées TLS 1.0, TLS 1.1 et/ou TLS 1.2 .

Vous pouvez définir cette variable d'environnement TLS dans le JCL de démarrage de l'initiateur de canal à l'aide de l'instruction de définition de données CEEOPTS:

```
CEEOPTS DD DSN=<dataset-name>,DISP=SHR
```

Dans le jeu de données référencé ci-dessus, spécifiez la liste que vous souhaitez utiliser, par exemple:

```
ENVAR("GSK_CLIENT_ECURVE_LIST=002300240025")
```

Important : N'utilisez pas cette instruction CEEOPTS avec des données de flux, car cela empêche la définition de la variable d'environnement pour toutes les tâches TLS utilisant cette instruction.

Veillez à référencer un fichier séquentiel ou un membre de fichier partitionné pour que cela fonctionne lors de l'utilisation d'une valeur SSLTASKS supérieure à 1.

Vous pouvez également utiliser l'équivalent analogique du serveur GSK_CLIENT_ECURVE_LIST, qui est GSK_SERVER_ALLOWED_KEX_ECURVES. Pour plus d'informations, voir Limitation des courbes elliptiques d'échange de clés .

En outre, voir le tableau 5 dans Définitions de la suite de chiffrement pour obtenir la liste des courbes elliptiques à 4 caractères valides et les spécifications des groupes pris en charge.

La spécification par défaut est 00210023002400250019. Si TLS V1.3 est activé, 0029 (x25519) est ajouté à la fin de la liste par défaut.

Identification et authentification des utilisateurs

Vous pouvez identifier et authentifier les utilisateurs à l'aide de certificats X.509 , de la structure MQCSP ou de plusieurs types de programme d'exit utilisateur.

Utilisation de certificats X.509

Vous pouvez identifier et authentifier les utilisateurs à l'aide de certificats x.509 avec la commande **CHLAUTH** et le paramètre **SSLPEER** . Le paramètre **SSLPEER** spécifie un filtre à utiliser pour la comparaison avec le nom distinctif du sujet du certificat provenant du gestionnaire de files d'attente ou du client homologue à l'autre extrémité du canal.

Pour plus d'informations sur l'utilisation de la commande **CHLAUTH** et du paramètre **SSLPEER** , voir SET CHLAUTH.

Utilisation de la structure MQCSP

Vous spécifiez la structure des paramètres de sécurité de connexion MQCSP sur un appel MQCONNX ; cette structure contient un ID utilisateur et un mot de passe. Si nécessaire, vous pouvez modifier le MQCSP dans un exit de sécurité.

Remarque : Le gestionnaire des droits d'accès aux objets (OAM) n'utilise pas le mot de passe. Cependant, l'OAM effectue un travail limité avec l'ID utilisateur, ce qui peut être considéré comme une forme d'authentification triviale. Ces vérifications vous empêchent d'adopter un autre ID utilisateur, si vous utilisez ces paramètres dans vos applications.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont protégés de manière appropriée, voir «Protection par mot de passe MQCSP», à la page 30.

Implémentation de l'identification et de l'authentification dans les exits de sécurité

L'objectif principal d'un exit de sécurité est d'activer l'agent MCA à chaque extrémité d'un canal pour authentifier son partenaire. A chaque extrémité d'un canal de transmission de messages et à l'extrémité serveur d'un canal MQI, un agent MCA agit généralement pour le compte du gestionnaire de files d'attente auquel il est connecté. A l'extrémité client d'un canal MQI, un agent MCA agit généralement pour le compte de l'utilisateur de l'application client IBM MQ . Dans cette situation, l'authentification mutuelle a lieu entre deux gestionnaires de files d'attente ou entre un gestionnaire de files d'attente et l'utilisateur d'une application IBM MQ MQI client .

L'exit de sécurité fourni (l'exit de canal SSPI) illustre comment l'authentification mutuelle peut être implémentée en échangeant des jetons d'authentification qui sont générés, puis vérifiés, par un serveur d'authentification sécurisé tel que Kerberos. Pour plus de détails, voir «Programme d'exit de canal SSPI sous Windows», à la page 155.

L'authentification mutuelle peut également être mise en oeuvre à l'aide de la technologie PKI (Public Key Infrastructure). Chaque exit de sécurité génère des données aléatoires, les signe à l'aide de la clé privée du gestionnaire de files d'attente ou de l'utilisateur qu'il représente et envoie les données signées à son partenaire dans un message de sécurité. L'exit de sécurité partenaire effectue l'authentification en vérifiant la signature numérique à l'aide de la clé publique du gestionnaire de files d'attente ou de l'utilisateur. Avant d'échanger des signatures numériques, les exits de sécurité peuvent avoir besoin de convenir de l'algorithme de génération d'un résumé de message, si plusieurs algorithmes sont disponibles pour être utilisés.

Lorsqu'un exit de sécurité envoie les données signées à son partenaire, il doit également envoyer un moyen d'identifier le gestionnaire de files d'attente ou l'utilisateur qu'il représente. Il peut s'agir d'un nom distinctif ou même d'un certificat numérique. Si un certificat numérique est envoyé, l'exit de sécurité partenaire peut le valider en utilisant la chaîne de certificats pour le certificat de l'autorité de certification racine. Cela garantit la propriété de la clé publique utilisée pour vérifier la signature numérique.

L'exit de sécurité partenaire ne peut valider un certificat numérique que s'il a accès à un référentiel de clés qui contient les certificats restants dans la chaîne de certificats. Si un certificat numérique pour le gestionnaire de files d'attente ou l'utilisateur n'est pas envoyé, il doit être disponible dans le référentiel de clés auquel l'exit de sécurité partenaire a accès. L'exit de sécurité partenaire ne peut pas vérifier la signature numérique sauf s'il peut trouver la clé publique du signataire.

Le protocole TLS (Transport Layer Security) utilise des techniques PKI telles que celles qui viennent d'être décrites. Pour plus d'informations sur la manière dont TLS effectue l'authentification, voir «Concepts TLS (Transport Layer Security)», à la page 15.

Si un serveur d'authentification sécurisé ou une prise en charge de l'infrastructure PKI n'est pas disponible, d'autres techniques peuvent être utilisées. Une technique commune, qui peut être implémentée dans les exits de sécurité, utilise un algorithme de clé symétrique.

L'un des exits de sécurité, l'exit A, génère un nombre aléatoire et l'envoie dans un message de sécurité à son exit de sécurité partenaire, l'exit B. L'exit B chiffre le nombre à l'aide de sa copie d'une clé connue uniquement des deux exits de sécurité. L'exit B envoie le numéro chiffré à l'exit A dans un message de sécurité avec un deuxième nombre aléatoire que l'exit B a généré. L'exit A vérifie que le premier nombre aléatoire a été chiffré correctement, chiffre le deuxième nombre aléatoire à l'aide de sa copie de la clé et envoie le nombre chiffré à l'exit B dans un message de sécurité. La sortie B vérifie alors que le deuxième nombre aléatoire a été chiffré correctement. Lors de cet échange, si l'un des exits de sécurité n'est pas satisfait de l'authenticité d'un autre, il peut demander à l'agent MCA de fermer le canal.

Un avantage de cette technique est qu'aucune clé ou mot de passe n'est envoyé sur la connexion de communication lors de l'échange. Un inconvénient est qu'il ne permet pas de résoudre le problème de la répartition sécurisée de la clé partagée. Une solution à ce problème est décrite dans «Implémentation de la confidentialité dans les programmes d'exit utilisateur», à la page 462. Une technique similaire est utilisée dans SNA pour l'authentification mutuelle de deux unités logiques lorsqu'elles se lient pour former une session. Cette technique est décrite dans «Authentification au niveau de la session», à la page 120.

Toutes les techniques d'authentification mutuelle précédentes peuvent être adaptées pour fournir une authentification unidirectionnelle.

Implémentation de l'identification et de l'authentification dans les exits de message

Lorsqu'une application place un message dans une file d'attente, la zone *UserIdentifier* du descripteur de message contient un ID utilisateur associé à l'application. Toutefois, il n'existe aucune donnée pouvant être utilisée pour authentifier l'ID utilisateur. Ces données peuvent être ajoutées par un exit de message à l'extrémité émettrice d'un canal et vérifiées par un exit de message à l'extrémité réceptrice du canal. Les données d'authentification peuvent être par example un mot de passe chiffré ou une signature numérique.

Ce service peut être plus efficace s'il est implémenté au niveau de l'application. La condition de base est que l'utilisateur de l'application qui reçoit le message puisse identifier et authentifier l'utilisateur de l'application qui a envoyé le message. Il est donc naturel d'envisager la mise en oeuvre de ce service au niveau de l'application. Pour plus d'informations, voir «Mappage d'identité dans l'exit d'API et l'exit de croisement d'API», à la page 346.

Implémentation de l'identification et de l'authentification dans l'exit d'API et l'exit de croisement d'API

Au niveau d'un message individuel, l'identification et l'authentification sont un service qui implique deux utilisateurs, l'expéditeur et le destinataire du message. La condition de base est que l'utilisateur de l'application qui reçoit le message puisse identifier et authentifier l'utilisateur de l'application qui a envoyé le message. Notez que l'exigence concerne l'authentification unidirectionnelle et non bidirectionnelle.

Selon la façon dont il est implémenté, les utilisateurs et leurs applications peuvent avoir besoin d'interfacer, voire d'interagir, avec le service. En outre, le moment et la manière dont le service est utilisé peuvent dépendre de l'emplacement des utilisateurs et de leurs applications, ainsi que de la nature

des applications elles-mêmes. Il est donc naturel d'envisager d'implémenter le service au niveau de l'application plutôt qu'au niveau de la liaison.

Si vous envisagez d'implémenter ce service au niveau de la liaison, vous devrez peut-être résoudre les problèmes suivants:

- Sur un canal de transmission de messages, comment appliquer le service uniquement aux messages qui en ont besoin?
- Comment autorisez-vous les utilisateurs et leurs applications à interagir avec le service, si c'est une exigence?
- Dans une situation à plusieurs tronçons, où un message est envoyé via plusieurs canaux de transmission de messages sur le chemin de sa destination, où appelez-vous les composants du service?

Voici quelques exemples de la façon dont le service d'identification et d'authentification peut être implémenté au niveau de l'application. Le terme *exit d'API* signifie un exit d'API ou un exit de croisement d'API.

- Lorsqu'une application place un message dans une file d'attente, un exit API peut acquérir un jeton d'authentification à partir d'un serveur d'authentification sécurisé tel que Kerberos. L'exit API peut ajouter ce jeton aux données d'application dans le message. Lorsque le message est extrait par l'application de réception, un deuxième exit API peut demander au serveur d'authentification d'authentifier l'expéditeur en vérifiant le jeton.
- Lorsqu'une application place un message dans une file d'attente, un exit API peut ajouter les éléments suivants aux données d'application du message:
  - Certificat numérique de l'expéditeur
  - Signature numérique de l'expéditeur

  Si des algorithmes différents sont disponibles pour la génération d'un résumé de message, l'exit d'API peut inclure le nom de l'algorithme qu'il a utilisé.

  Lorsque le message est extrait par l'application de réception, un deuxième exit d'API peut effectuer les vérifications suivantes:

  - L'exit API peut valider le certificat numérique en utilisant la chaîne de certificats pour le certificat de l'autorité de certification racine. Pour ce faire, l'exit API doit avoir accès à un référentiel de clés qui contient les certificats restants dans la chaîne de certificats. Cette vérification garantit que l'expéditeur, identifié par le nom distinctif, est le véritable propriétaire de la clé publique contenue dans le certificat.
  - L'exit API peut vérifier la signature numérique à l'aide de la clé publique contenue dans le certificat. Cette vérification authentifie l'expéditeur.

  Le nom distinctif de l'expéditeur peut être envoyé à la place du certificat numérique complet. Dans ce cas, le référentiel de clés doit contenir le certificat de l'expéditeur pour que le deuxième exit d'API puisse trouver la clé publique de l'expéditeur. Une autre possibilité consiste à envoyer tous les certificats de la chaîne de certificats.
- Lorsqu'une application place un message dans une file d'attente, la zone *UserIdentifier* du descripteur de message contient un ID utilisateur associé à l'application. L'ID utilisateur peut être utilisé pour identifier l'expéditeur. Pour activer l'authentification, un exit d'API peut ajouter des données, telles qu'un mot de passe chiffré, aux données d'application du message. Lorsque le message est extrait par l'application de réception, un deuxième exit API peut authentifier l'ID utilisateur à l'aide des données qui ont été transmises avec le message.

  Cette technique peut être considérée comme suffisante pour les messages provenant d'un environnement contrôlé et sécurisé, et dans les cas où un serveur d'authentification sécurisé ou une prise en charge de l'infrastructure PKI n'est pas disponible.

Méthode PAM (Pluggable Authentication Method)

Le module PAM est désormais commun aux plateformes UNIX and Linux et fournit un mécanisme général qui masque les détails de l'authentification des utilisateurs des services.

Des règles d'authentification différentes peuvent être utilisées pour différents services, en configurant les règles, sans qu'aucune modification ne soit nécessaire pour les services eux-mêmes.

Pour plus d'informations, voir «Utilisation de la méthode PAM (Pluggable Authentication Method)», à la page 360.

Utilisateurs privilégiés

Un utilisateur privilégié est un utilisateur disposant de droits d'administration complets pour IBM MQ.

Outre les utilisateurs répertoriés dans le tableau suivant, il existe certains objets et autorisations pour lesquels une attention particulière doit être accordée lors de l'octroi de l'accès, afin de garantir l'intégrité et la sécurité du gestionnaire de files d'attente. Un examen supplémentaire doit être effectué lors de l'octroi de l'une des autorisations suivantes:

- Toute autorisation sur les objets SYSTEM
- Autorisations d'administration pour créer, modifier et supprimer des objets.

  z/OS Sous z/OS, cette autorisation correspond aux droits de sécurité de la commande et de la ressource de commande permettant d'exécuter les commandes DEFINE, ALTER et DELETE.

  Multi Sur toutes les autres plateformes, ces autorisations sont des autorisations d'administration telles que `+crt`, `+chg` et `+dlt`.
- Autorisation d'administration pour effacer les files d'attente.

  z/OS Sous z/OS, cette autorisation correspond à la sécurité des commandes et aux droits de sécurité des ressources de commandes permettant d'émettre des commandes CLEAR.

  Multi Sur toutes les autres plateformes, cette autorisation est `+clr`.
- Autorisations d'administration permettant d'arrêter des canaux, d'annuler ou de valider des messages.

  z/OS Sous z/OS, cette autorisation est une autorisation de sécurité de commande et de sécurité de ressource de commande permettant d'émettre des commandes telles que RESET CHANNEL, START CHANNEL et STOP CHANNEL.

  Multi Sur toutes les autres plateformes, ces autorisations sont `+ctrl` et `+ctrlx`.
- Autorisation MQI de l'utilisateur de remplacement qui permet aux applications d'augmenter les privilèges pour les vérifications d'autorisation.

  z/OS Sous z/OS, cette autorisation correspond à tout droit accordé aux autres profils de sécurité utilisateur.

  Multi Sur toutes les autres plateformes, cette autorisation est `+altusr`.
- Autorisations de contexte qui permettent aux applications de modifier le contexte de sécurité des messages.

  z/OS Sous z/OS, cette autorisation correspond à toute autorisation accordée aux profils de sécurité de contexte.

  Multi Sur toutes les autres plateformes, ces autorisations sont `+setall` et `+setid`.

En tant que principe général, les applications de messagerie ne doivent recevoir que les autorisations MQI de base pour les files d'attente ou les rubriques nécessaires. Les canaux MCA qui s'exécutent sous un MCAUSER non privilégié et certains autres types d'applications spéciaux, tels que les gestionnaires de files d'attente de rebut, peuvent nécessiter des autorisations supplémentaires qui ne sont normalement pas accordées aux applications pour fonctionner correctement.

Tableau 67. Utilisateurs privilégiés par plateforme	
Plateforme	**Utilisateurs privilégiés**
Systèmes Windows	• SYSTEME • Membres du groupe mqm • Membres du groupe Administrateurs
Systèmes UNIX and Linux	• Membres du groupe mqm
IBM i IBM i Systèmes IBM i	• Les profils qmqm et qmqmadm • Tous les membres du groupe qmqmadm • Tout utilisateur défini avec le paramètre *ALLOBJ
z/OS	ID utilisateur sous lequel s'exécutent les espaces adresse de l'initiateur de canal, du gestionnaire de files d'attente et de la sécurité avancée des messages. Ces ID utilisateur ne disposent pas automatiquement de droits d'administration complets pour IBM MQ, mais ils sont considérés comme privilégiés en raison du niveau d'accès généralement accordé à ces ID utilisateur.

Identification et authentification des utilisateurs à l'aide de la structure MQCSP

Vous pouvez spécifier la structure des paramètres de sécurité de connexion MQCSP sur un appel MQCONNX.

La structure des paramètres de sécurité de connexion MQCSP contient un ID utilisateur et un mot de passe que le service d'autorisation peut utiliser pour identifier et authentifier l'utilisateur.

Vous pouvez modifier le MQCSP dans un exit de sécurité.

Avertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont protégés de manière appropriée, voir «Protection par mot de passe MQCSP», à la page 30.

Relation entre les paramètres MQCSP et AdoptCTX

IBM MQ authentifie toujours les données d'identification transmises via la structure MQCSP, sauf si la fonction d'authentification de connexion n'est pas activée. Une fois que les données d'identification ont été authentifiées avec succès, IBM MQ tente d'adopter l'ID utilisateur pour les futures vérifications d'autorisation, sauf si ADOPTCTX n'est pas activé.

IBM MQ a une limite sur la longueur des ID utilisateur qu'il peut utiliser pour les vérifications d'autorisation. Ces limites sont détaillées dans «ID utilisateur», à la page 84. Lors de l'adoption d'un ID utilisateur transmis via la structure MQCSP, IBM MQ se comporte différemment, en fonction des autres options de configuration:

- Lors de l'utilisation de l'authentification de connexion LDAP, IBM MQ extrait la valeur de la zone définie dans SHORTUSR à partir de l'enregistrement LDAP de l'utilisateur de cet utilisateur et adopte cet ID utilisateur.

  Par exemple, si SHORTUSR est défini sur `'CN'` et qu'un enregistrement LDAP répertorie un utilisateur en tant que `'CN=Test,SN=MQ,O=IBM,C=UK'`, l'ID utilisateur `Test` est utilisé.
- Lors de l'utilisation de l'authentification de connexion au système d'exploitation ou de l'authentification PAM, si ADOPTCTX est défini sur YES, l'ID utilisateur transmis via la structure MQCSP est tronqué afin

de respecter la limite d'ID utilisateur de 12 caractères de IBM MQ lorsqu'il est adopté comme contexte de connexion.

Si **ChlAuthEarlyAdopt** est activé, la troncature se produit une fois que les données d'identification de l'utilisateur ont été authentifiées.

Si **ChlAuthEarlyAdopt** n'est pas activé, la troncature est effectuée avant l'adoption. Sous Windows, si l'utilisateur est indiqué au format `user@domain`, cela signifie que la troncature peut entraîner une spécification de domaine qui n'est pas valide lorsque l'utilisateur comporte moins de 12 caractères.

Par exemple, si un utilisateur `` `ibmmq@windowsdomain` `` est fourni via MQCSP, il est tronqué à `` `ibmmq@window` `` dans ce scénario. Il en résulte l'erreur suivante:

```
AMQ8074W: L'autorisation a échoué car le SID'SID'ne correspond pas à l'entité'ibmmq@window'
```

Dans ce cas, si vous transmettez un ID utilisateur de plus de 12 caractères, tel qu'un ID utilisateur de domaine Windows au format `user@domain`, via le MQCSP, vous devez configurer **ChlAuthEarlyAdopt**=*Y* dans le fichier qm.ini pour éviter cette erreur.

Vous pouvez également utiliser ADOPTCTX (NO) dans la configuration CONNAUTH AUTHINFO et utiliser une autre approche, telle qu'une règle CHLAUTH USERMAP, un exit de sécurité ou le paramètre MCAUSER de l'objet de canal pour définir l'ID utilisateur du canal.

Implémentation de l'identification et de l'authentification dans les exits de sécurité

Vous pouvez utiliser un exit de sécurité pour implémenter l'authentification unidirectionnelle ou mutuelle.

L'objectif principal d'un exit de sécurité est d'activer l'agent MCA à chaque extrémité d'un canal pour authentifier son partenaire. A chaque extrémité d'un canal de transmission de messages et à l'extrémité serveur d'un canal MQI, un agent MCA agit généralement pour le compte du gestionnaire de files d'attente auquel il est connecté. A l'extrémité client d'un canal MQI, un agent MCA agit généralement pour le compte de l'utilisateur de l'application IBM MQ MQI client . Dans cette situation, l'authentification mutuelle a lieu entre deux gestionnaires de files d'attente ou entre un gestionnaire de files d'attente et l'utilisateur d'une application IBM MQ MQI client .

L'exit de sécurité fourni (l'exit de canal SSPI) illustre comment l'authentification mutuelle peut être implémentée en échangeant des jetons d'authentification qui sont générés, puis vérifiés, par un serveur d'authentification sécurisé tel que Kerberos. Pour plus de détails, voir «Programme d'exit de canal SSPI sous Windows», à la page 155.

L'authentification mutuelle peut également être mise en oeuvre à l'aide de la technologie PKI (Public Key Infrastructure). Chaque exit de sécurité génère des données aléatoires, les signe à l'aide de la clé privée du gestionnaire de files d'attente ou de l'utilisateur qu'il représente et envoie les données signées à son partenaire dans un message de sécurité. L'exit de sécurité partenaire effectue l'authentification en vérifiant la signature numérique à l'aide de la clé publique du gestionnaire de files d'attente ou de l'utilisateur. Avant d'échanger des signatures numériques, les exits de sécurité peuvent avoir besoin de convenir de l'algorithme de génération d'un résumé de message, si plusieurs algorithmes sont disponibles pour être utilisés.

Lorsqu'un exit de sécurité envoie les données signées à son partenaire, il doit également envoyer un moyen d'identifier le gestionnaire de files d'attente ou l'utilisateur qu'il représente. Il peut s'agir d'un nom distinctif ou même d'un certificat numérique. Si un certificat numérique est envoyé, l'exit de sécurité partenaire peut le valider en utilisant la chaîne de certificats pour le certificat de l'autorité de certification racine. Cela garantit la propriété de la clé publique utilisée pour vérifier la signature numérique.

L'exit de sécurité partenaire ne peut valider un certificat numérique que s'il a accès à un référentiel de clés qui contient les certificats restants dans la chaîne de certificats. Si un certificat numérique pour le gestionnaire de files d'attente ou l'utilisateur n'est pas envoyé, il doit être disponible dans le référentiel de clés auquel l'exit de sécurité partenaire a accès. L'exit de sécurité partenaire ne peut pas vérifier la signature numérique sauf s'il peut trouver la clé publique du signataire.

Le protocole TLS (Transport Layer Security) utilise des techniques PKI telles que celles qui viennent d'être décrites. Pour plus d'informations sur la manière dont la couche Secure Sockets Layer effectue l'authentification, voir «Concepts TLS (Transport Layer Security)», à la page 15.

Si un serveur d'authentification sécurisé ou une prise en charge de l'infrastructure PKI n'est pas disponible, d'autres techniques peuvent être utilisées. Une technique commune, qui peut être implémentée dans les exits de sécurité, utilise un algorithme de clé symétrique.

L'un des exits de sécurité, l'exit A, génère un nombre aléatoire et l'envoie dans un message de sécurité à son exit de sécurité partenaire, l'exit B. L'exit B chiffre le nombre à l'aide de sa copie d'une clé connue uniquement des deux exits de sécurité. L'exit B envoie le numéro chiffré à l'exit A dans un message de sécurité avec un deuxième nombre aléatoire que l'exit B a généré. L'exit A vérifie que le premier nombre aléatoire a été chiffré correctement, chiffre le deuxième nombre aléatoire à l'aide de sa copie de la clé et envoie le nombre chiffré à l'exit B dans un message de sécurité. La sortie B vérifie alors que le deuxième nombre aléatoire a été chiffré correctement. Lors de cet échange, si l'un des exits de sécurité n'est pas satisfait de l'authenticité d'un autre, il peut demander à l'agent MCA de fermer le canal.

Un avantage de cette technique est qu'aucune clé ou mot de passe n'est envoyé sur la connexion de communication lors de l'échange. Un inconvénient est qu'il ne permet pas de résoudre le problème de la répartition sécurisée de la clé partagée. Une solution à ce problème est décrite dans «Implémentation de la confidentialité dans les programmes d'exit utilisateur», à la page 462. Une technique similaire est utilisée dans SNA pour l'authentification mutuelle de deux unités logiques lorsqu'elles se lient pour former une session. Cette technique est décrite dans «Authentification au niveau de la session», à la page 120.

Toutes les techniques d'authentification mutuelle précédentes peuvent être adaptées pour fournir une authentification unidirectionnelle.

Mappage d'identité dans les exits de message

Vous pouvez utiliser des exits de message pour traiter des informations afin d'authentifier un ID utilisateur, mais il peut être préférable d'implémenter l'authentification au niveau de l'application.

Lorsqu'une application place un message dans une file d'attente, la zone *UserIdentifier* du descripteur de message contient un ID utilisateur associé à l'application. Toutefois, il n'existe aucune donnée pouvant être utilisée pour authentifier l'ID utilisateur. Ces données peuvent être ajoutées par un exit de message à l'extrémité émettrice d'un canal et vérifiées par un exit de message à l'extrémité réceptrice du canal. Les données d'authentification peuvent être par example un mot de passe chiffré ou une signature numérique.

Ce service peut être plus efficace s'il est implémenté au niveau de l'application. La condition de base est que l'utilisateur de l'application qui reçoit le message puisse identifier et authentifier l'utilisateur de l'application qui a envoyé le message. Il est donc naturel d'envisager la mise en oeuvre de ce service au niveau de l'application. Pour plus d'informations, voir «Mappage d'identité dans l'exit d'API et l'exit de croisement d'API», à la page 346.

Mappage d'identité dans l'exit d'API et l'exit de croisement d'API

Une application qui reçoit un message doit être en mesure d'identifier et d'authentifier l'utilisateur de l'application qui a envoyé le message. Ce service est généralement mieux implémenté au niveau de l'application. Les exits API peuvent implémenter le service de différentes manières.

Au niveau d'un message individuel, l'identification et l'authentification sont un service qui implique deux utilisateurs, l'expéditeur et le destinataire du message. La condition de base est que l'utilisateur de l'application qui reçoit le message puisse identifier et authentifier l'utilisateur de l'application qui a envoyé le message. Notez que l'exigence concerne l'authentification unidirectionnelle et non bidirectionnelle.

Selon la façon dont il est implémenté, les utilisateurs et leurs applications peuvent avoir besoin d'interfacer, voire d'interagir, avec le service. En outre, le moment et la manière dont le service est utilisé peuvent dépendre de l'emplacement des utilisateurs et de leurs applications, ainsi que de la nature

des applications elles-mêmes. Il est donc naturel d'envisager d'implémenter le service au niveau de l'application plutôt qu'au niveau de la liaison.

Si vous envisagez d'implémenter ce service au niveau de la liaison, vous devrez peut-être résoudre les problèmes suivants:

- Sur un canal de transmission de messages, comment appliquer le service uniquement aux messages qui en ont besoin?
- Comment autorisez-vous les utilisateurs et leurs applications à interagir avec le service, si c'est une exigence?
- Dans une situation à plusieurs tronçons, où un message est envoyé via plusieurs canaux de transmission de messages sur le chemin de sa destination, où appelez-vous les composants du service?

Voici quelques exemples de la façon dont le service d'identification et d'authentification peut être implémenté au niveau de l'application. Le terme *exit d'API* signifie un exit d'API ou un exit de croisement d'API.

- Lorsqu'une application place un message dans une file d'attente, un exit API peut acquérir un jeton d'authentification à partir d'un serveur d'authentification sécurisé tel que Kerberos. L'exit API peut ajouter ce jeton aux données d'application dans le message. Lorsque le message est extrait par l'application de réception, un deuxième exit API peut demander au serveur d'authentification d'authentifier l'expéditeur en vérifiant le jeton.
- Lorsqu'une application place un message dans une file d'attente, un exit API peut ajouter les éléments suivants aux données d'application du message:
  - Certificat numérique de l'expéditeur
  - Signature numérique de l'expéditeur

  Si des algorithmes différents sont disponibles pour la génération d'un résumé de message, l'exit d'API peut inclure le nom de l'algorithme qu'il a utilisé.

  Lorsque le message est extrait par l'application de réception, un deuxième exit d'API peut effectuer les vérifications suivantes:

  - L'exit API peut valider le certificat numérique en utilisant la chaîne de certificats pour le certificat de l'autorité de certification racine. Pour ce faire, l'exit API doit avoir accès à un référentiel de clés qui contient les certificats restants dans la chaîne de certificats. Cette vérification garantit que l'expéditeur, identifié par le nom distinctif, est le véritable propriétaire de la clé publique contenue dans le certificat.
  - L'exit API peut vérifier la signature numérique à l'aide de la clé publique contenue dans le certificat. Cette vérification authentifie l'expéditeur.

  Le nom distinctif de l'expéditeur peut être envoyé à la place du certificat numérique complet. Dans ce cas, le référentiel de clés doit contenir le certificat de l'expéditeur pour que le deuxième exit d'API puisse trouver la clé publique de l'expéditeur. Une autre possibilité consiste à envoyer tous les certificats de la chaîne de certificats.
- Lorsqu'une application place un message dans une file d'attente, la zone *UserIdentifier* du descripteur de message contient un ID utilisateur associé à l'application. L'ID utilisateur peut être utilisé pour identifier l'expéditeur. Pour activer l'authentification, un exit d'API peut ajouter des données, telles qu'un mot de passe chiffré, aux données d'application du message. Lorsque le message est extrait par l'application de réception, un deuxième exit API peut authentifier l'ID utilisateur à l'aide des données qui ont été transmises avec le message.

  Cette technique peut être considérée comme suffisante pour les messages provenant d'un environnement contrôlé et sécurisé, et dans les cas où un serveur d'authentification sécurisé ou une prise en charge de l'infrastructure PKI n'est pas disponible.

Utilisation des certificats révoqués

Les certificats numériques peuvent être révoqués par les autorités de certification. Vous pouvez vérifier le statut de révocation des certificats à l'aide d'OCSP ou de CRL sur les serveurs LDAP, en fonction de la plateforme.

Lors de l'établissement de liaison TLS, les partenaires communicants s'authentifient mutuellement avec des certificats numériques. L'authentification peut inclure une vérification du certificat reçu. Les autorités de certification révoquent les certificats pour diverses raisons, notamment:

- Le propriétaire a été déplacé vers une autre organisation
- La clé privée n'est plus un secret

Les autorités de certification publient les certificats personnels révoqués dans une liste de révocation de certificat (CRL). Les certificats d'autorités de certification qui ont été révoqués sont publiés dans une liste de révocation des droits d'accès (ARL).

Sur les plateformes suivantes, la prise en charge SSL de IBM MQ recherche les certificats révoqués à l'aide du protocole OCSP (Online Certificate Status Protocol) ou des listes CRL et ARL sur les serveurs LDAP (Lightweight Directory Access Protocol). OCSP est la méthode préférée.

- 

Linux
- 

UNIX
- 

Windows

IBM MQ classes for Java et IBM MQ classes for JMS ne peuvent pas utiliser les informations OCSP dans un fichier de table de définition de canal du client. Toutefois, vous pouvez configurer OCSP comme indiqué dans Using Online Certificate Protocol.

Sur les plateformes suivantes, et la prise en charge SSL de IBM MQ vérifie les certificats révoqués à l'aide de listes CRL et ARL sur les serveurs LDAP uniquement:

- 

IBM i
- 

z/OS

Pour plus d'informations sur les autorités de certification, voir «Certificats numériques», à la page 9.

Vérification d'OCSP/CRL

La vérification du protocole OCSP (Online Certificate Status Protocol) /de la liste de révocation de certificat (CRL) est effectuée par rapport aux certificats entrants distants. Le processus vérifie l'ensemble de la chaîne impliquée depuis le certificat personnel du système distant jusqu'à son certificat racine.

Utilisation d' openSSL pour vérifier la validation OCSP

Si votre entreprise utilise openSSL pour valider OCSP, puis que vous tentez d'utiliser une connexion GSKit TLS, vous recevez un avertissement de statut UNKNOWN.

En effet, tous les certificats de la chaîne, à l'exception de la racine, sont vérifiés par GSKit pour le statut de révocation. L'opération GSKit est conforme à la RFC 5280 et est décrite dans la stratégie GSKit Trust. L'algorithme GSKit essaie toutes les sources disponibles pour les informations de révocation, comme décrit dans RFC 5280 et dans la stratégie GSKit Trust.

Comment la vérification OCSP/CRL fonctionne-t-elle dans IBM MQ?

IBM MQ prend en charge deux mécanismes permettant de contrôler le comportement lors de la vérification des certificats par rapport aux noeuds finaux OCSP ou CRL nommés, soit dans l'extension de certificat, soit, comme défini dans les objets AUTHINFO:

- Les attributs **OCSPCheckExtensions**, **CDPCheckExtensions**et **OCSPAuthentication** de la strophe SSL du fichier qm.ini, et
- Utilisation du paramètre SSLCRLNL du gestionnaire de files d'attente et des configurations AUTHINFO OCSP et CRLLDAP. Pour plus d'informations, voir ALTER AUTHINFO et ALTER QMGR .

Avertissement :

La commande ALTER AUTHINFO avec **AUTHTYPE(OCSP)** ne s'applique pas aux gestionnaires de files d'attente IBM i ou z/OS . Toutefois, il peut être spécifié sur ces plateformes pour être copié dans la table de définition de canal du client (CCDT) à des fins d'utilisation par le client.

Les attributs de strophe SSL **OCSPCheckExtensions** et **CDPCheckExtensions** contrôlent si IBM MQ doit vérifier un certificat par rapport au serveur OCSP ou CRL détaillé dans l'extension AIA du certificat.

Si cette option n'est pas activée, le serveur OCSP ou CRL de l'extension de certificat n'est pas contacté.

Si des serveurs OCSP ou CRL sont détaillés via des objets AUTHINFO et référencés à l'aide de l'attribut SSLCRLNL **QMGR** , lors du traitement de la révocation de certificat, IBM MQ tente de contacter ces serveurs.

Important : Un seul objet OCSP AUTHINFO peut être défini dans la liste de noms SSLCRLNL.

If :

OCSPCheckExtensions= NO et **CDPCheckExtensions**=*NO* sont définis, et
Aucun serveur OCSP ou CRL n'est défini dans les objets AUTHINFO

aucune vérification de révocation de certificat n'est effectuée.

Lors de la vérification d'un certificat pour son statut de révocation, IBM MQ contacte les serveurs OCSP ou CRL nommés dans l'ordre suivant, s'ils sont activés:

1. Le serveur OCSP est détaillé dans un objet **AUTHTYPE(OCSP)** et référencé dans l'attribut SSLCRLNL **QMGR** .
2. Serveurs OCSP détaillés dans l'extension AIA des certificats, si **OCSPCheckExtensions**=*YES*.
3. Serveurs CRL détaillés dans l'extension **CRLDistributionPoints** des certificats, si **CDPCheckExtensions** =*YES*.
4. Tous les serveurs CRL détaillés dans les objets **AUTHINFO(CRLLDAP)** et référencés dans l'attribut SSLCRLNL **QMGR** .

Lors de la vérification d'un certificat, si une étape aboutit à ce que le serveur OCSP ou le serveur CRL renvoie une réponse REVOKED ou VALID définitive à une requête pour le certificat, aucune autre vérification n'est effectuée et le statut du certificat tel que présenté est utilisé pour déterminer s'il est digne de confiance ou non.

Si un serveur OCSP ou CRL renvoie un résultat UNKNOWN, le traitement se poursuit jusqu'à ce qu'un serveur OCSP ou CRL renvoie un résultat définitif ou que toutes les options soient épuisées.

Le comportement selon lequel un certificat est considéré comme révoqué, si son statut ne peut pas être déterminé, est différent pour les serveurs OCSP et CRL:

- Pour les serveurs CRL, si aucune CRL ne peut être obtenue, le certificat est considéré comme NOT_REVOKED
- Pour les serveurs OCSP, si aucun statut de révocation ne peut être obtenu à partir d'un serveur OCSP nommé, le comportement est contrôlé via l'attribut **OCSPAuthentication** dans la strophe SSL du fichier qm.ini .

  Vous pouvez configurer cet attribut pour bloquer une connexion, autoriser une connexion ou autoriser une connexion avec un message d'avertissement.

Vous pouvez utiliser l'attribut **SSLHTTPProxyName**=*string* dans la strophe SSL des fichiers qm.ini et mqclient.ini pour les vérifications OCSP, si nécessaire. La chaîne correspond au nom d'hôte ou à l'adresse réseau du serveur proxy HTTP qui doit être utilisé par GSKit pour les vérifications OCSP.

Depuis la IBM MQ 9.1.5 , vous pouvez définir la valeur **OCSPTimeout** dans la strophe SSL des fichiers `qm.ini` ou `mqclient.ini` qui définit le nombre de secondes d'attente d'un répondeur OCSP lors de l'exécution d'une vérification de révocation.

ULW Certificats révoqués et OCSP

IBM MQ détermine le répondeur OCSP (Online Certificate Status Protocol) à utiliser et traite la réponse reçue. Vous pouvez être amené à réaliser certaines étapes pour pouvoir accéder au répondeur OCSP.

Remarque : Ces informations s'appliquent uniquement à IBM MQ sur les systèmes UNIX, Linux, and Windows .

Pour vérifier le statut de révocation d'un certificat numérique à l'aide d'OCSP, IBM MQ peut utiliser deux méthodes pour déterminer le répondeur OCSP à contacter:

- A l'aide de l'extension de certificat AuthorityInfoAccess (AIA) dans le certificat à vérifier.
- A l'aide d'une adresse URL spécifiée dans un objet d'informations d'authentification ou spécifiée par une application client.

Une URL spécifiée dans un objet d'informations d'authentification ou par une application client est prioritaire par rapport à une URL d'une extension de certificat AIA.

Si l'adresse URL du répondeur OCSP se trouve derrière le pare-feu, reconfigurez le pare-feu pour que le répondeur OCSP soit accessible ou configurez un serveur proxy OCSP. Indiquez le nom du serveur proxy en utilisant la variable SSLHTTPProxyName dans la strophe SSL. Sur les systèmes client, vous pouvez également indiquer le nom du serveur proxy à l'aide de la variable d'environnement MQSSLPROXY. Pour plus de détails, consultez les informations connexes.

Si vous n'êtes pas concerné par la révocation des certificats TLS, peut-être parce que vous exécutez un environnement de test, vous pouvez définir OCSPCheckExtensions sur NO dans la strophe SSL. Si vous configurez cette variable, toute extension de certificat AIA est ignorée. Cette solution sera probablement refusée dans un environnement de production, dans lequel vous ne souhaitez sûrement pas autoriser les utilisateurs à accéder aux certificats révoqués.

L'appel d'accès au répondeur OCSP peut entraîner l'un des trois résultats suivants :

Bon
Le certificat est valide.

Révoqué
Le certificat est révoqué.

Inconnu
Ce résultat peut survenir à cause de l'une des trois raisons suivantes :

- IBM MQ ne peut pas accéder au répondeur OCSP.
- Le répondeur OCSP a envoyé une réponse, mais IBM MQ ne peut pas vérifier la signature numérique de la réponse.
- Le répondeur OCSP a envoyé une réponse qui indique qu'il n'existe pas de données de révocation pour le certificat.

Si IBM MQ reçoit un résultat OCSP `Inconnu`, son comportement dépend de la valeur de l'attribut OCSPAuthentication. Pour les gestionnaires de files d'attente, cet attribut est conservé dans l'un des emplacements suivants:

- Linux UNIX Dans la section SSL du fichier `qm.ini` sous UNIX and Linux.
- Windows Dans le registre Windows .

Cet attribut peut être défini à l'aide de IBM MQ Explorer. Pour les clients, l'attribut est conservé dans la section SSL du fichier de configuration du client.

Si `Inconnu` est reçu et que l'attribut OCSPAuthentication a la valeur REQUIRED (valeur par défaut), IBM MQ rejette la connexion et envoie un message d'erreur de type AMQ9716.

Si les messages d'événements SSL de gestionnaire de files d'attente sont activés, un message d'événement SSL de type MQRC_CHANNEL_SSL_ERROR, avec ReasonQualifier défini sur MQRQ_SSL_HANDSHAKE_ERROR, est généré.

Si `Inconnu` est reçu et que l'attribut OCSPAuthentication a la valeur OPTIONAL, IBM MQ permet au canal SSL de démarrer, et aucun avertissement ou message d'événement SSL n'est généré.

Si `Inconnu` est reçu et que l'attribut OCSPAuthentication a la valeur WARN, le canal SSL démarre, mais IBM MQ génère un message d'avertissement de type AMQ9717 dans le journal des erreurs. Si les messages d'événements SSL de gestionnaire de files d'attente sont activés, un message d'événement SSL de type MQRC_CHANNEL_SSL_WARNING, avec ReasonQualifier défini sur MQRQ_SSL_UNKNOWN_REVOCATION, est généré.

Signature numérique de réponses OCSP

Un répondeur OCSP peut signer ses réponses de trois manières. Votre répondeur vous informe de la méthode à utiliser.

- La réponse OCSP peut être signée numériquement à l'aide du même certificat CA qui a émis le certificat en cours de vérification. Dans ce cas, vous n'avez pas besoin de configurer de certificat supplémentaire ; les étapes que vous avez déjà effectuées pour établir la connectivité TLS sont suffisantes pour vérifier la réponse OCSP.
- La réponse OCSP peut être signée numériquement à l'aide d'un autre certificat signé par la même autorité de certification que celle ayant émis le certificat en cours de vérification. Le certificat signataire est envoyé avec la réponse OCSP dans ce cas. Le certificat transmis à partir du répondeur OCSP doit avoir une extension d'utilisation clé étendue définie sur `id-kp-OCSPSigning` pour pouvoir être digne de confiance. Etant donné que la réponse OCSP est envoyée avec le certificat qui l'a signé (et que ce certificat est signé par une autorité de certification déjà digne de confiance pour la connectivité TLS), aucune configuration de certificat supplémentaire n'est requise.
- La réponse OCSP peut être signée numériquement à l'aide d'un autre certificat qui n'est pas directement lié au certificat en cours de vérification. Dans ce cas, la réponse OCSP est signée par un certificat émis par le répondeur OCSP. Vous devez ajouter une copie du certificat de répondeur OCSP à la base de données de clés du client ou du gestionnaire de files d'attente qui effectue la vérification OCSP ; voir «Ajout d'un certificat de l'autorité de certification ou de la partie publique d'un certificat autosigné dans un référentiel de clés sur UNIX, Linux, and Windows», à la page 314 . Lors de l'ajout d'un certificat CA, il est ajouté par défaut en racine de confiance, ce qui représente le paramètre requis dans ce contexte. Si ce certificat n'est pas ajouté, IBM MQ ne peut pas vérifier la signature numérique sur la réponse OCSP et la vérification OCSP génère un résultat Inconnu, ce qui peut entraîner la fermeture du canal par IBM MQ , en fonction de la valeur d'OCSPAuthentication.

Protocole OCSP (Online Certificate Status Protocol) dans les applications client Java et JMS

En raison d'une limitation de l'API Java , IBM MQ peut utiliser la vérification de révocation de certificat OCSP (Online Certificate Status Protocol) pour les sockets TLS sécurisés uniquement lorsque OCSP est activé pour l'ensemble du processus de la machine virtuelle Java (JVM). OCSP peut être activé pour toutes les connexions sécurisées de la machine virtuelle Java de deux manières :

- En modifiant le fichier JRE java.security pour y inclure les paramètres de configuration OCSP affichés dans le tableau 1 et en redémarrant l'application.
- Utilisez java.security.Security.setProperty() API, soumise à toute stratégie Java Security Manager en vigueur.

Vous devez au moins spécifier l'une des valeurs ocsp.enable et ocsp.responderURL.

Nom de la propriété	Description
ocsp.enable	Cette propriété a la valeur `true` ou `false`. Si la valeur est `true`, la vérification OCSP est activée lors de la vérification de révocation de

Nom de la propriété	Description
	certificat. Si la valeur est `false` (ou non définie), la vérification OCSP est désactivée.
ocsp.responderURL	La valeur de cette propriété est une adresse URL identifiant l'emplacement du canal répondeur OCSP. Par exemple, `ocsp.responderURL=http://ocsp.example.net:80`. Par défaut, l'emplacement du canal répondeur OCSP est déterminé de manière implicite à partir du certificat en cours de validation. La propriété est utilisée lorsque l'extension Authority Information Access (définie dans RFC 3280) n'est pas indiquée dans le certificat ou lorsqu'elle doit être remplacée.
ocsp.responderCertSubjectName	La valeur de cette propriété est le nom du sujet du certificat du canal répondeur OCSP. Par exemple, `ocsp.responderCertSubjectName="CN=OCSP Responder, O=XYZ Corp"`. Par défaut, le certificat du canal répondeur OCSP est celui de l'émetteur du certificat en cours de validation. Cette propriété identifie le certificat du canal répondeur OCSP lorsque la valeur par défaut ne s'applique pas. Sa valeur est un nom distinctif de chaîne (défini dans RFC 2253) qui identifie un certificat dans l'ensemble des certificats fournis lors de la validation du chemin aux certificats. Lorsque le seul nom du sujet ne suffit pas à identifier le certificat, alors les propriétés ocsp.responderCertIssuerName et ocsp.responderCertSerialNumber doivent toutes deux être utilisées. Lorsque cette propriété est définie, les propriétés ocsp.responderCertIssuerName et ocsp.responderCertSerialNumber sont ignorées.
ocsp.responderCertIssuerName	La valeur de cette propriété est le nom de l'émetteur du certificat du canal répondeur OCSP. Par exemple, `ocsp.responderCertIssuerName="CN=Enterprise CA, O=XYZ Corp"`. Par défaut, le certificat du canal répondeur OCSP est celui de l'émetteur du certificat en cours de validation. Cette propriété identifie le certificat du canal répondeur OCSP lorsque la valeur par défaut ne s'applique pas. Sa valeur est un nom distinctif de chaîne (défini dans RFC 2253) qui identifie un certificat dans l'ensemble des certificats fournis lors de la validation du chemin aux certificats. Lorsque cette propriété est définie, la propriété ocsp.responderCertSerialNumber doit également être définie. Cette propriété est ignorée lorsque la propriété ocsp.responderCertSubjectName est définie.
ocsp.responderCertSerialNumber	La valeur de cette propriété est le numéro de série du certificat du canal répondeur OCSP. Par exemple, `ocsp.responderCertSerialNumber=2A:FF:00`. Par défaut, le certificat du canal répondeur OCSP est celui de l'émetteur du certificat en cours de validation. Cette propriété identifie le certificat du canal répondeur OCSP lorsque la valeur par défaut ne s'applique pas. Cette valeur est une chaîne de chiffres hexadécimaux (séparés par des signes deux-points ou des espaces) qui identifie un certificat dans l'ensemble des certificats fournis lors de la validation du chemin d'accès aux certificats. Lorsque cette propriété est définie, la propriété ocsp.responderCertIssuerName doit également être définie. Cette propriété est ignorée lorsque la propriété ocsp.responderCertSubjectName est définie.

Avant d'activer OCSP de cette manière, tenez compte des remarques suivantes :

- La définition de la configuration OCSP affecte toutes les connexions sécurisées du processus de la machine virtuelle Java. Dans certains cas, cette configuration peut avoir des effets secondaires indésirables lorsque la machine virtuelle Java est partagée avec un autre code d'application qui utilise des sockets TLS sécurisés. Assurez-vous que la configuration OCSP choisie est appropriée à toutes les applications s'exécutant sur la même machine virtuelle Java.
- L'application de la maintenance à votre environnement d'exécution Java écrase le fichier java.security. Soyez prudent lorsque vous appliquez des correctifs temporaires Java et la maintenance du produit pour éviter d'écraser le fichier java.security . Il peut s'avérer nécessaire d'appliquer à nouveau les modifications de votre fichier java.security après la maintenance. Pour cette raison, vous pouvez envisager de définir la configuration OCSP à l'aide de l'interface de programme d'application java.security.Security.setProperty().
- L'activation de la vérification OCSP n'est effective que si la vérification de la révocation est également activée. La vérification de la révocation est activée via la méthode `PKIXParameters.setRevocationEnabled()`.
- Si vous utilisez l'intercepteur AMS Java décrit dans Activation de la vérification OCSP dans les intercepteurs natifs, prenez soin d'éviter d'utiliser une configuration OCSP java.security qui entre en conflit avec la configuration AMS OCSP dans le fichier de configuration du magasin de clés.

Utilisation des listes de révocation de certificat et des listes de révocation d'autorité

La prise en charge de IBM MQ pour les CRL et les ARL varie en fonction de la plateforme.

Le support CRL et ARL sur chaque plateforme est le suivant:

- Sous z/OS, System SSL prend en charge les listes CRL et ARL stockées sur les serveurs LDAP par le produit Tivoli Public Key Infrastructure.
- Sur les autres plateformes, la prise en charge de CRL et ARL est conforme aux recommandations de profil de CRL PKIX X.509 V2 .

IBM MQ gère un cache des listes de révocation de certificat et des listes de révocation de certificat qui ont été consultées au cours des 12 dernières heures.

Lorsqu'un gestionnaire de files d'attente ou IBM MQ MQI client reçoit un certificat, il vérifie la liste de révocation de certificat pour confirmer que le certificat est toujours valide. IBM MQ vérifie d'abord dans le cache s'il existe un cache. Si la liste de révocation de certificat n'est pas dans le cache, IBM MQ interroge les emplacements du serveur de listes de révocation de certificat LDAP dans l'ordre dans lequel ils apparaissent dans la liste de noms des objets d'informations d'authentification spécifiée par l'attribut *SSLCRLNL* , jusqu'à ce que IBM MQ trouve une liste de révocation de certificat disponible. Si la liste de noms n'est pas spécifiée ou qu'elle est spécifiée avec une valeur vide, les listes de révocation de nom ne sont pas vérifiées.

Configuration des serveurs LDAP

Configurez la structure de l'arborescence d'informations de l'annuaire LDAP pour refléter la hiérarchie des noms distinctifs des autorités de certification. Pour ce faire, utilisez les fichiers LDAP Data Interchange Format.

Configurez la structure DIT (Directory Information Tree) LDAP pour utiliser la hiérarchie correspondant aux noms distinctifs des autorités de certification qui émettent les certificats et les CRL. Vous pouvez configurer la structure DIT avec un fichier qui utilise le format LDIF (LDAP Data Interchange Format). Vous pouvez également utiliser des fichiers LDIF pour mettre à jour un répertoire.

Les fichiers LDIF sont des fichiers texte ASCII qui contiennent les informations requises pour définir des objets dans un annuaire LDAP. Les fichiers LDIF contiennent une ou plusieurs entrées, dont chacune comprend un nom distinctif, au moins une définition de classe d'objet et, éventuellement, plusieurs définitions d'attribut.

L'attribut `certificateRevocationList;binary` contient une liste, au format binaire, des certificats d'utilisateur révoqués. L'attribut `authorityRevocationList;binary` contient une liste binaire des certificats de l'autorité de certification qui ont été révoqués. Pour une utilisation avec IBM MQ TLS, les

données binaires de ces attributs doivent être conformes au format DER (Défini Encoding Rules). Pour plus d'informations sur les fichiers LDIF, reportez-vous à la documentation fournie avec votre serveur LDAP.

La Figure 20, à la page 354 présente un exemple de fichier LDIF que vous pouvez créer en entrée de votre serveur LDAP pour charger les CRL et les ARL émis par CA1, qui est une autorité de certification imaginaire avec le nom distinctif "CN=CA1, OU=Test, O=IBM, C=GB", configuré par l'organisation de test dans IBM.

```
dn: o=IBM, c=GB
o: IBM
objectclass: top
objectclass: organization

dn: ou=Test, o=IBM, c=GB
ou: Test
objectclass: organizationalUnit

dn: cn=CA1, ou=Test, o=IBM, c=GB
cn: CA1
objectclass: cRLDistributionPoint
objectclass: certificateAuthority
authorityRevocationList;binary:: (DER format data)
certificateRevocationList;binary:: (DER format data)
caCertificate;binary:: (DER format data)
```

Figure 20. Exemple de fichier LDIF pour une autorité de certification. Cela peut varier d'une implémentation à l'autre.

La Figure 21, à la page 354 montre la structure DIT créée par votre serveur LDAP lorsque vous chargez l'exemple de fichier LDIF présenté dans Figure 20, à la page 354 avec un fichier similaire pour CA2, une autorité de certification imaginaire configurée par l'organisation PKI, également dans IBM.

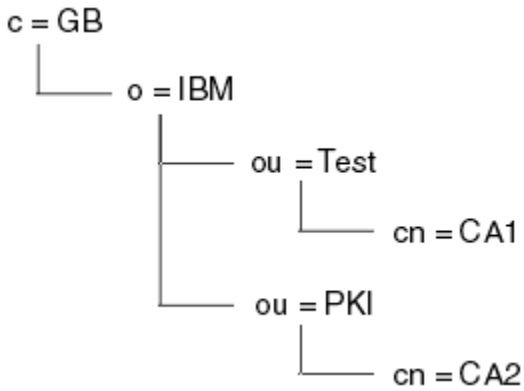


Figure 21. Exemple de structure d'arborescence d'informations d'annuaire LDAP

WebSphere MQ vérifie à la fois les CRL et les ARL.

Remarque : Assurez-vous que la liste de contrôle d'accès de votre serveur LDAP permet aux utilisateurs autorisés de lire, de rechercher et de comparer les entrées qui contiennent les CRL et les ARL. WebSphere MQ accède au serveur LDAP à l'aide des propriétés LDAPUSER et LDAPPWD de l'objet AUTHINFO.

Configuration et mise à jour des serveurs LDAP
Utilisez cette procédure pour configurer ou mettre à jour votre serveur LDAP.

1. Procurez-vous les listes de révocation de certificat et les listes de révocation de certificat au format DER auprès de votre ou de vos autorités de certification.
2. A l'aide d'un éditeur de texte ou de l'outil fourni avec votre serveur LDAP, créez un ou plusieurs fichiers LDIF contenant le nom distinctif de l'autorité de certification et les définitions de classe d'objets requises. Copiez les données au format DER dans le fichier LDIF en tant

que valeurs de l'attribut `certificateRevocationList;binary` pour les CRL, de l'attribut `authorityRevocationList;binary` pour les ARL, ou les deux.

3. Démarrez votre serveur LDAP.
4. Ajoutez les entrées du ou des fichiers LDIF que vous avez créés à l'étape «2», à la page 354.

Après avoir configuré votre serveur CRL LDAP, vérifiez qu'il est correctement configuré. Tout d'abord, essayez d'utiliser un certificat qui n'est pas révoqué sur le canal et vérifiez que le canal démarre correctement. Utilisez ensuite un certificat révoqué et vérifiez que le canal ne démarre pas.

Obtenir fréquemment des listes de révocation de certificats mises à jour auprès des autorités de certification. Envisagez de le faire sur vos serveurs LDAP toutes les 12 heures.

Accès aux CRL et aux ARL à l'aide d'un gestionnaire de files d'attente

Un gestionnaire de files d'attente est associé à un ou plusieurs objets d'informations d'authentification, qui contiennent l'adresse d'un serveur CRL LDAP. IBM i IBM MQ on IBM i se comporte différemment des autres plateformes.

Notez que dans cette section, les informations sur les listes de révocation de certificat (CRL) s'appliquent également aux listes de révocation d'autorité (ARL).

Vous indiquez au gestionnaire de files d'attente comment accéder aux listes de révocation de certificat en lui fournissant des objets d'informations d'authentification, chacun contenant l'adresse d'un serveur de listes de révocation de certificat LDAP. Les objets d'informations d'authentification sont conservés dans une liste de noms, qui est spécifiée dans l'attribut de gestionnaire de files d'attente *SSLCRLNL* .

Dans l'exemple suivant, MQSC est utilisé pour spécifier les paramètres:

1. Définissez les objets d'informations d'authentification à l'aide de la commande DEFINE AUTHINFO MQSC, avec le paramètre AUTHTYPE défini sur CRLLDAP. IBM i Sous IBM i, vous pouvez également utiliser la commande CL CRTMQMAUTI.

   La valeur CRLLDAP pour le paramètre AUTHTYPE indique que les CRL sont accessibles sur les serveurs LDAP. Chaque objet d'informations d'authentification de type CRLLDAP que vous créez contient l'adresse d'un serveur LDAP. Lorsque vous disposez de plusieurs objets d'informations d'authentification, les serveurs LDAP vers lesquels ils pointent doivent contenir des informations identiques. Cela assure la continuité du service en cas d'échec d'un ou de plusieurs serveurs LDAP.

   z/OS De plus, sous z/OS uniquement, tous les serveurs LDAP doivent être accessibles à l'aide des mêmes ID utilisateur et mot de passe. L'ID utilisateur et le mot de passe utilisés sont ceux indiqués dans le premier objet AUTHINFO de la liste de noms.

   Sur toutes les plateformes, l'ID utilisateur et le mot de passe sont envoyés au serveur LDAP en clair.
2. A l'aide de la commande DEFINE NAMELIST MQSC, définissez une liste de noms pour les noms de vos objets d'informations d'authentification. z/OS Sous z/OS, vérifiez que l'attribut de liste de noms NLTYPE est défini sur AUTHINFO.
3. A l'aide de la commande ALTER QMGR MQSC, fournissez la liste de noms au gestionnaire de files d'attente. Exemple :

   ```
   ALTER QMGR SSLCRLNL(sslcrlnlname)
   ```

   où `sslcrlnlname` est votre liste de noms d'objets d'informations d'authentification.

   Cette commande définit un attribut de gestionnaire de files d'attente appelé *SSLCRLNL*. La valeur initiale du gestionnaire de files d'attente pour cet attribut est vide.

IBM i Sous IBM i, vous pouvez spécifier des objets d'informations d'authentification, mais le gestionnaire de files d'attente n'utilise ni des objets d'informations d'authentification ni une liste de noms d'objets d'informations d'authentification. Seuls les clients IBM MQ qui utilisent une table de connexion client générée par un gestionnaire de files d'attente IBM i utilisent les informations d'authentification

spécifiées pour ce gestionnaire de files d'attente IBM i . L'attribut de gestionnaire de files d'attente *SSLCRLNL* dans IBM i détermine les informations d'authentification utilisées par ces clients. Pour savoir comment indiquer à un gestionnaire de files d'attente IBM i comment accéder aux listes de révocation de certificat, voir «Accès aux CRL et aux ARL sous IBM i», à la page 356 .

Vous pouvez ajouter jusqu'à 10 connexions à des serveurs LDAP de remplacement à la liste de noms afin d'assurer la continuité du service en cas d'échec d'un ou de plusieurs serveurs LDAP. Notez que les serveurs LDAP doivent contenir des informations identiques.

IBM i *Accès aux CRL et aux ARL sous IBM i*

Utilisez cette procédure pour accéder aux CRL ou aux ARL sous IBM i.

Notez que dans cette section, les informations sur les listes de révocation de certificat (CRL) s'appliquent également aux listes de révocation d'autorité (ARL).

Pour configurer un emplacement de liste de révocation de certificat pour un certificat spécifique sur IBM i, procédez comme suit:

1. Accédez à l'interface DCM, comme décrit dans «Accès à DCM», à la page 282.
2. Dans la catégorie de tâche **Gérer les emplacements de liste de révocation de certificat** du panneau de navigation, cliquez sur **Ajouter un emplacement de liste de révocation de certificat**. La page Gérer les emplacements de liste de révocation de certificat s'affiche dans le cadre de la tâche.
3. Dans la zone **Nom d'emplacement de liste de révocation de certificat** , entrez un nom d'emplacement de liste de révocation de certificat, par exemple LDAP `Server #1`
4. Dans la zone **Serveur LDAP** , entrez le nom du serveur LDAP.
5. Dans la zone **Utiliser SSL (Secure Sockets Layer)** , sélectionnez **Oui** si vous souhaitez vous connecter au serveur LDAP à l'aide de TLS. Sinon, sélectionnez **Non**.
6. Dans la zone **Numéro de port** , entrez un numéro de port pour le serveur LDAP, par exemple 389.
7. Si votre serveur LDAP n'autorise pas les utilisateurs anonymes à interroger l'annuaire, entrez un nom distinctif de connexion pour le serveur dans la zone **Nom distinctif de connexion** .
8. Cliquez sur **OK**. DCM vous informe qu'il a créé l'emplacement de la liste de révocation de certificat.
9. Dans le panneau de navigation, cliquez sur **Sélectionner un magasin de certificats**. La page Sélectionner un magasin de certificats s'affiche dans le cadre de la tâche.
10. Cochez la case **Autre magasin de certificats système** et cliquez sur **Continuer**. La page Magasin de certificats et mot de passe s'affiche.
11. Dans la zone **Chemin d'accès au magasin de certificats et nom de fichier** , entrez le chemin d'accès au système de fichiers intégré et le nom de fichier que vous définissez lorsque «Création d'un magasin de certificats sous IBM i», à la page 283.
12. Entrez un mot de passe dans la zone **Certificate Store Password** . Cliquez sur **Continuer**. La page Magasin de certificats en cours s'affiche dans le cadre de la tâche.
13. Dans la catégorie de tâche **Gérer les certificats** du panneau de navigation, cliquez sur **Mettre à jour l'affectation d'emplacement de liste de révocation de certificat**. La page Affectation d'emplacement de liste de révocation de certificat s'affiche dans le cadre de la tâche.
14. Sélectionnez le bouton d'option du certificat de l'autorité de certification auquel vous souhaitez affecter l'emplacement de la liste de révocation de certificat. Cliquez sur **Mettre à jour l'affectation d'emplacement de liste de révocation de certificat**. La page Mettre à jour l'affectation d'emplacement de liste de révocation de certificat s'affiche dans le cadre de la tâche.
15. Sélectionnez le bouton d'option correspondant à l'emplacement de la liste de révocation de certificat que vous souhaitez affecter au certificat. Cliquez sur **Mettre à jour l'affectation**. DCM vous informe qu'il a mis à jour l'affectation.

Notez que DCM vous permet d'affecter un serveur LDAP différent par l'autorité de certification.

Accès aux CRL et aux ARL à l'aide de IBM MQ Explorer
Vous pouvez utiliser IBM MQ Explorer pour indiquer à un gestionnaire de files d'attente comment accéder aux CRL.

Notez que dans cette section, les informations sur les listes de révocation de certificat (CRL) s'appliquent également aux listes de révocation d'autorité (ARL).

Utilisez la procédure suivante pour configurer une connexion LDAP à une CRL:

1. Vérifiez que vous avez démarré votre gestionnaire de files d'attente.
2. Cliquez avec le bouton droit de la souris sur le dossier **Informations d'authentification** , puis cliquez sur **Nouveau-> Informations d'authentification**. Dans la feuille de propriétés qui s'ouvre:
   a. Sur la première page **Créer des informations d'authentification**, entrez un nom pour l'objet CRL (LDAP).
   b. Dans la page **Général** de **Modifier les propriétés**, sélectionnez le type de connexion. Vous pouvez éventuellement entrer une description.
   c. Sélectionnez la page **CRL (LDAP)** de **Modifier les propriétés**.
   d. Entrez le nom du serveur LDAP en tant que nom de réseau ou adresse IP.
   e. Si le serveur requiert des détails de connexion, indiquez un ID utilisateur et, si nécessaire, un mot de passe.
   f. Cliquez sur **OK**.
3. Cliquez avec le bouton droit de la souris sur le dossier Listes de noms , puis cliquez sur **Nouveau-> Liste de noms**. Dans la feuille de propriétés qui s'ouvre:
   a. Entrez un nom pour la liste de noms.
   b. Ajoutez le nom de l'objet CRL (LDAP) (à l'étape «2.a», à la page 357) à la liste.
   c. Cliquez sur **OK**.
4. Cliquez avec le bouton droit de la souris sur le gestionnaire de files d'attente, sélectionnez **Propriétés**, puis sélectionnez la page **SSL** :
   a. Cochez la case **Vérifier les certificats reçus par ce gestionnaire de files d'attente par rapport aux listes de révocation de certification** .
   b. Entrez le nom de la liste de noms (à l'étape «3.a», à la page 357) dans la zone **Liste de noms CRL** .

Accès aux CRL et aux ARL à l'aide d'un IBM MQ MQI client
Vous disposez de trois options pour spécifier les serveurs LDAP qui contiennent des listes CRL à vérifier par un IBM MQ MQI client.

Notez que dans cette section, les informations sur les listes de révocation de certificat (CRL) s'appliquent également aux listes de révocation d'autorité (ARL).

Les trois méthodes de spécification des serveurs LDAP sont les suivantes:

- Utilisation d'une table de définition de canal
- Utilisation de la structure des options de configuration SSL, MQSCO, sur un appel MQCONNX
- Utilisation de Active Directory (sur les systèmes Windows avec prise en charge d' Active Directory)

Pour plus de détails, reportez-vous aux informations associées.

Vous pouvez inclure jusqu'à 10 connexions à d'autres serveurs LDAP pour assurer la continuité du service en cas d'échec d'un ou de plusieurs serveurs LDAP. Notez que les serveurs LDAP doivent contenir des informations identiques.

Vous ne pouvez pas accéder aux CRL LDAP à partir d'un canal IBM MQ MQI client s'exécutant sur Linux (plateforme zSeries).

Emplacement d'un répondeur OCSP et des serveurs LDAP qui contiennent des listes CRL
Sur un système IBM MQ MQI client , vous pouvez spécifier l'emplacement d'un répondeur OCSP et des serveurs LDAP (Lightweight Directory Access Protocol) qui contiennent des listes de révocation de certificat (CRL).

Vous pouvez spécifier ces emplacements de trois manières, décrites ici par ordre de priorité décroissante.

IBM i Pour IBM i, voir Accès aux CRL et aux ARL sur IBM i.

Lorsqu'une application IBM MQ MQI client émet un appel MQCONNX

Vous pouvez spécifier un répondeur OCSP ou un serveur LDAP contenant des CRL sur un appel **MQCONNX** .

Sur un appel **MQCONNX** , la structure d'options de connexion, MQCNO, peut faire référence à une structure d'options de configuration SSL, MQSCO. A son tour, la structure MQSCO peut référencer une ou plusieurs structures d'enregistrement d'informations d'authentification, MQAIR. Chaque structure MQAIR contient toutes les informations dont IBM MQ MQI client a besoin pour accéder à un répondeur OCSP ou à un serveur LDAP contenant des CRL. Par exemple, l'une des zones d'une structure MQAIR est l'URL à laquelle un répondeur peut être contacté. Pour plus d'informations sur la structure MQAIR, voir MQAIR-Enregistrement des informations d'authentification.

Utilisation d'une table de définition de canal du client (ccdt) pour accéder à un répondeur OCSP ou à des serveurs LDAP

Pour qu'un IBM MQ MQI client puisse accéder à un répondeur OCSP ou à des serveurs LDAP qui contiennent des CRL, incluez les attributs d'un ou de plusieurs objets d'informations d'authentification dans une table de définition de canal du client.

Sur un gestionnaire de files d'attente de serveur, vous pouvez définir un ou plusieurs objets d'informations d'authentification. Les attributs d'un objet d'authentification contiennent toutes les informations requises pour accéder à un répondeur OCSP (sur les plateformes où OCSP est pris en charge) ou à un serveur LDAP qui contient des CRL. L'un des attributs spécifie l'URL du répondeur OCSP, l'autre l'adresse de l'hôte ou l'adresse IP d'un système sur lequel s'exécute un serveur LDAP.

z/OS IBM i Un objet d'informations d'authentification avec AUTHTYPE (OCSP) ne s'applique pas aux gestionnaires de files d'attente IBM i ou z/OS , mais il peut être spécifié sur ces plateformes pour être copié dans la table de définition de canal du client (CCDT) à des fins d'utilisation par le client.

Pour permettre à un IBM MQ MQI client d'accéder à un répondeur OCSP ou à des serveurs LDAP qui contiennent des CRL, les attributs d'un ou de plusieurs objets d'informations d'authentification peuvent être inclus dans une table de définition de canal du client. Vous pouvez inclure ces attributs de l'une des manières suivantes:

Multi

Sur les plateformes serveur AIX, Linux, IBM i, Solariset Windows

Vous pouvez définir une liste de noms contenant les noms d'un ou de plusieurs objets d'informations d'authentification. Vous pouvez ensuite définir l'attribut de gestionnaire de files d'attente, **SSLCRLNL**, sur le nom de cette liste de noms.

Si vous utilisez des listes de révocation de certificat, plusieurs serveurs LDAP peuvent être configurés pour fournir une disponibilité plus élevée. L'objectif est que chaque serveur LDAP dispose des mêmes CRL. Si un serveur LDAP n'est pas disponible lorsqu'il est requis, un IBM MQ MQI client peut tenter d'accéder à un autre serveur.

Les attributs des objets d'informations d'authentification identifiés par la liste de noms sont appelés collectivement ici *emplacement de révocation de certificat*. Lorsque vous définissez l'attribut de gestionnaire de files d'attente, **SSLCRLNL**, sur le nom de la liste de noms, l'emplacement de révocation de certificat est copié dans la table de définition de canal du client associée au gestionnaire de files d'attente. Si la table de définition de canal du client est accessible à partir d'un

système client en tant que fichier partagé, ou si la table de définition de canal du client est ensuite copiée sur un système client, le IBM MQ MQI client sur ce système peut utiliser l'emplacement de révocation de certificat dans la table de définition de canal du client pour accéder à un répondeur OCSP ou à des serveurs LDAP contenant des listes de révocation de certificat.

Si l'emplacement de révocation de certificat du gestionnaire de files d'attente est modifié ultérieurement, la modification est reflétée dans la table de définition de canal du client associée au gestionnaire de files d'attente. Si l'attribut de gestionnaire de files d'attente, **SSLCRLNL**, est mis à blanc, l'emplacement de révocation de certificat est supprimé de la table de définition de canal du client. Ces modifications ne sont reflétées dans aucune copie de la table sur un système client.

Si vous souhaitez que l'emplacement de révocation de certificat aux extrémités client et serveur d'un canal MQI soit différent et que le gestionnaire de files d'attente du serveur est celui qui est utilisé pour créer l'emplacement de révocation de certificat, procédez comme suit:

1. Sur le gestionnaire de files d'attente du serveur, créez l'emplacement de révocation de certificat à utiliser sur le système client.
2. Copiez la table de définition de canal du client contenant l'emplacement de révocation de certificat sur le système client.
3. Sur le gestionnaire de files d'attente du serveur, remplacez l'emplacement de révocation de certificat par l'emplacement requis à l'extrémité serveur du canal MQI.
4. Sur la machine client, vous pouvez utiliser la commande **runmqsc** avec le paramètre **-n** .

Multi

Sur les plateformes client AIX, Linux, IBM i , Solariset Windows

Vous pouvez générer une table de définition de canal du client sur la machine client à l'aide de la commande runmqsc avec le paramètre **-n** et les objets **DEFINE AUTHINFO** du fichier CCDT. L'ordre dans lequel les objets sont définis est l'ordre dans lequel ils sont utilisés dans le fichier. Tout nom que vous pouvez utiliser dans un objet **DEFINE AUTHINFO** n'est pas conservé dans le fichier. Seuls les nombres à position fixe sont utilisés lorsque vous **DISPLAY** les objets **AUTHINFO** dans un fichier CCDT.

Remarque : Si vous spécifiez le paramètre **-n** , vous ne devez spécifier aucun autre paramètre.

Utilisation d' Active Directory sous Windows

Windows

Sur les systèmes Windows , vous pouvez utiliser la commande de contrôle **setmqcrl** pour publier les informations CRL en cours dans Active Directory.

La commande **setmqcrl** ne publie pas les informations OCSP.

Pour plus d'informations sur cette commande et sa syntaxe, voir setmqcrl.

Accès aux CRL et aux ARL avec IBM MQ classes for Java et IBM MQ classes for JMS

IBM MQ classes for Java et IBM MQ classes for JMS accèdent aux CRL différemment des autres plateformes.

Pour plus d'informations sur l'utilisation des CRL et des ARL avec IBM MQ classes for Java, voir Utilisation des listes de révocation de certificat

Pour plus d'informations sur l'utilisation des CRL et des ARL avec IBM MQ classes for JMS, voir Propriété d'objet SSLCERTSTORES

Manipulation des objets d'informations d'authentification

Vous pouvez manipuler des objets d'informations d'authentification à l'aide de commandes MQSC ou PCF ou du IBM MQ Explorer.

Les commandes MQSC suivantes agissent sur les objets d'informations d'authentification:

- DEFINE AUTHINFO
- ALTER AUTHINFO
- DELETE AUTHINFO
- INFORMATIONS D'AUTHENTIFICATION D'AFFICHAGE

Pour une description complète de ces commandes, voir Commandes MQSC.

Les commandes PCF (Programmable Command Format) suivantes agissent sur les objets d'informations d'authentification:

- Créer des informations d'authentification
- Copier des informations d'authentification
- Modifier des informations d'authentification
- Supprimer des informations d'authentification
- Consulter des informations d'authentification
- Consulter des noms d'informations d'authentification

Pour une description complète de ces commandes, voir Définitions des formats de commande programmables.

Sur les plateformes où il est disponible, vous pouvez également utiliser le IBM MQ Explorer.

Linux UNIX Utilisation de la méthode PAM (Pluggable Authentication Method)

Vous pouvez utiliser PAM uniquement sur les plateformes UNIX and Linux . Un système UNIX standard comporte des modules PAM qui implémentent le mécanisme d'authentification traditionnel ; toutefois, il peut y en avoir d'autres. En plus de la tâche de base de validation des mots de passe, les modules PAM peuvent également être appelés pour exécuter des règles supplémentaires.

Les fichiers de configuration définissent la méthode d'authentification à utiliser pour chaque application. Les exemples d'application incluent la connexion de terminal standard, ftp et telnet.

L'avantage de PAM est que l'application n'a pas besoin de connaître ou de se soucier de la façon dont l'ID utilisateur est authentifié. Tant que l'application peut fournir une forme correcte de données d'authentification à PAM, le mécanisme qui la sous-tend est transparent.

La forme des données d'authentification dépend du système utilisé. Par exemple, IBM MQ obtient un mot de passe via des paramètres, tels que la structure MQCSP utilisée dans l'appel API `MQCONNX` .

Important : Vous ne pouvez pas définir l'attribut **AUTHENMD** tant que vous n'avez pas installé IBM MQ 8.0.0 Fix Pack 3, puis redémarré le gestionnaire de files d'attente à l'aide d'un niveau **-e CMDLEVEL=** de *802* (dans la commande strmqm) pour définir le niveau de commande requis.

Configuration de votre système pour l'utilisation de PAM

Le nom de service utilisé par IBM MQlors de l'appel de PAM est *ibmmq*.

Notez qu'une installation IBM MQ tente de gérer une configuration PAM par défaut, qui autorise les connexions des utilisateurs du système d'exploitation, en fonction des valeurs par défaut connues pour les différents systèmes d'exploitation.

Toutefois, votre administrateur système doit vérifier que les règles définies dans les fichiers `/etc/pam.conf`ou `/etc/pam.d/ibmmq`sont toujours appropriées.

Autorisation de l'accès aux objets

Cette section contient des informations sur l'utilisation du gestionnaire de droits d'accès aux objets et des programmes d'exit de canal pour contrôler l'accès aux objets.

ULW Sur les systèmes UNIX, Linux, and Windows . vous contrôlez l'accès aux objets à l'aide du gestionnaire des droits d'accès aux objets (OAM). Cette collection de rubriques contient des informations sur l'utilisation de l'interface de commande de la méthode d'accès aux objets (OAM).

Cette section contient également une liste de contrôle que vous pouvez utiliser pour déterminer les tâches à effectuer pour appliquer la sécurité à votre système sur toutes les plateformes, ainsi que les considérations à prendre en compte pour accorder aux utilisateurs le droit d'administrer IBM MQ et d'utiliser les objets IBM MQ .

Si les mécanismes de sécurité fournis ne répondent pas à vos besoins, vous pouvez développer vos propres programmes d'exit de canal.

Identification de l'utilisateur utilisé pour l'autorisation

Les droits d'accès aux ressources sont accordés aux groupes dont l'utilisateur est membre ou, dans certains modes, directement à l'utilisateur associé à la connexion. Lors du processus de connexion, et en particulier pour les connexions distantes (client), cette identité peut être modifiée par la configuration du gestionnaire de files d'attente. Cette page répertorie les différentes fonctions d' IBM MQ et leurs options de configuration qui peuvent avoir un impact sur l'identité d'une application de connexion, ainsi que l'ordre de priorité dans lequel ces fonctions sont appliquées.

Fonctions pouvant modifier l'utilisateur adopté

Les différentes fonctions qui peuvent définir l'utilisateur qui doit être autorisé sont les suivantes:

Utilisateur vérifié par l'application
Lorsqu'une connexion distante est démarrée par IBM MQ, l'utilisateur du système d'exploitation sous lequel le processus s'exécute est envoyé au gestionnaire de files d'attente de réception. Cet utilisateur est envoyé pour s'assurer que s'il n'existe aucune configuration supplémentaire qui modifie l'utilisateur, il existe un utilisateur qui peut être utilisé pour la vérification des autorisations.

Il n'est pas recommandé d'utiliser cet utilisateur comme base d'autorisation car il permet aux connexions d'affirmer leur identité sans validation côté serveur. Cela peut même inclure l'administrateur ('mqm').

Paramètre MCAUSER du canal
Les applications qui se connectent via des liaisons réseau le font à l'aide d'une définition de canal IBM MQ . Les définitions de canal prennent en charge l'attribut **MCAUSER** , qui peut être utilisé pour spécifier un utilisateur différent à utiliser pour l'autorisation au lieu de l'utilisateur vérifié par les applications de connexion.

Authentification de connexion ADOPTCTX
Les applications peuvent spécifier un utilisateur et un mot de passe à envoyer à un gestionnaire de files d'attente à des fins d'authentification. Ces données d'identification sont authentifiées à l'aide de la configuration spécifiée pour la fonction d'authentification de connexion. L'option **ADOPTCTX** pour l'authentification de connexion contrôle si un utilisateur doit être utilisé pour l'autorisation après sa validation. Si la valeur est YES, l'utilisateur fourni pour l'authentification est adopté pour les vérifications d'autorisation.

Enregistrement d'authentification de canal MCAUSER
Lors du traitement de la connexion, le gestionnaire de files d'attente tente de trouver un enregistrement d'authentification de canal correspondant à la connexion. Si un enregistrement d'authentification de canal est mis en correspondance et que sa valeur d'attribut **USERSRC** est définie sur MAP, IBM MQ remplace l'utilisateur utilisé pour les autorisations par la valeur de l'attribut **MCAUSER** .

Exits de sécurité
Les exits de sécurité sont des fonctions personnalisées qui peuvent être écrites et appelées lors du traitement de la sécurité IBM MQ . Lorsque la fonction est appelée, elle est fournie avec une copie de la structure MQCD qui inclut plusieurs zones relatives à l'utilisateur des connexions qui seront utilisées pour les vérifications d'autorisation. Les exits de sécurité peuvent modifier ces zones pour modifier l'utilisateur qui sera autorisé.

ordre de précédence

Le tableau suivant présente l'ordre de priorité de chaque fonction de sécurité décrite dans «Fonctions pouvant modifier l'utilisateur adopté», à la page 361 lorsque IBM MQ sélectionne un utilisateur à autoriser. L'ordre est du plus bas au plus élevé, c'est-à-dire qu'une fonction de sécurité définissant un utilisateur à la première ligne est remplacée par l'une des autres lignes.

Tableau 68. Ordre de priorité des fonctions de sécurité

Commande	**Fonction**
1 (moins important)	ID vérifié par l'application
2	Définition de canal **MCAUSER** , attribut
3	Authentification de la connexion avec **ADOPTCTX(YES)**
4	Enregistrements d'authentification de canal avec **USERSRC(MAP)**
5 (plus élevé)	Exit de sécurité

Implications de l'adoption précoce

Les enregistrements d'authentification de connexion et d'authentification de canal fournissent une option de configuration qui contrôle le moment où l'adoption de l'utilisateur pour l'authentification de connexion est effectuée. Ce paramètre est appelé "adoption précoce". Si l'adoption anticipée est activée, l'adoption de l'identité d'authentification de connexion a lieu avant le traitement des enregistrements d'authentification de canal (ce qui signifie que les enregistrements d'authentification de canal remplacent toute adoption **CONNAUTH**).

Si cette option est désactivée, l'ordre est inversé, c'est-à-dire que les enregistrements d'authentification de canal sont traités avant l'adoption de **CONNAUTH** . Dans cette situation, l'adoption de l'authentification de connexion a une priorité effective plus élevée que les enregistrements d'authentification de canal.

Le paramètre par défaut pour l'adoption anticipée est `activé`.

ULW Contrôle de l'accès aux objets à l'aide de la méthode d'accès aux objets (OAM) sous UNIX, Linux, and Windows

Le gestionnaire des droits d'accès aux objets (OAM) fournit une interface de commande pour l'octroi et la révocation des droits d'accès aux objets IBM MQ .

Vous devez disposer des droits appropriés pour utiliser ces commandes, comme décrit dans «Droit d'administration de IBM MQ sur UNIX, Linux, and Windows», à la page 416. Les ID utilisateur autorisés à administrer IBM MQ disposent des droits *superutilisateur* sur le gestionnaire de files d'attente, ce qui signifie que vous n'avez pas besoin de leur accorder d'autres droits pour émettre des demandes ou des commandes MQI.

Linux UNIX Droits utilisateur OAM sur UNIX and Linux

Depuis IBM MQ 8.0, sur les systèmes UNIX and Linux , le gestionnaire des droits d'accès aux objets (OAM) peut utiliser l'autorisation basée sur l'utilisateur ainsi que l'autorisation basée sur le groupe.

Avant la IBM MQ 8.0, les listes de contrôle d'accès sous UNIX and Linux reposent uniquement sur des groupes. Depuis la IBM MQ 8.0, elles reposent sur les ID utilisateur et les groupes et vous pouvez utiliser le modèle qui s'appuie sur les utilisateurs ou le modèle qui s'appuie sur les groupes pour l'autorisation en associant l'attribut **SecurityPolicy** à la valeur appropriée comme décrit dans Configuring installable services et Configuring authorization service stanzas on UNIX and Linux.

Changements de comportement pour IBM MQ 8.0 et versions ultérieures

Depuis IBM MQ 8.0, lors de l'exécution avec la règle basée sur l'utilisateur, certaines commandes renvoient des informations différentes de celles des versions antérieures du produit:

- Les commandes **dmpmqaut** et **dmpmqcfg** affichent les enregistrements utilisateur, comme le opérations PCF équivalentes.
- Le plug-in OAM pour IBM MQ Explorer indique les enregistrements basés sur l'utilisateur et autorise les modifications basées sur l'utilisateur.
- La fonction **Inquire** d'OAM renvoie des résultats indiquant qu'elle est peut fonctionner en étant basée sur l'utilisateur.

L'utilisation de l'attribut **-p** dans la commande **setmqaut** n'accorde pas l'accès à tous les utilisateurs du même groupe principal, lorsque les autorisations basées sur les utilisateurs sont activées dans le fichier `qm.ini` , comme décrit dans la section Service du fichier qm.ini.

Si vous commencez à utiliser l'autorisation basée sur l'utilisateur et que vous disposez de nombreux utilisateurs, il y aura probablement davantage d'enregistrements stockés sur la file d'attente AUTH qu'avec le modèle basé sur le groupe et le processus d'autorisation risque de durer un peu plus longtemps qu'auparavant car les enregistrements à vérifier sont plus nombreux. Cette augmentation ne devrait pas être significative. Si nécessaire, vous pouvez utiliser une combinaison de droits basés sur l'utilisateur et le groupe.

Migration

Si vous modifiez le modèle du groupe à l'utilisateur pour un gestionnaire de files d'attente existant, l'effet n'est pas immédiat. Les autorisations ayant déjà été accordées continuent de s'appliquer. N'importe quel utilisateur se connectant au gestionnaire de files d'attente reçoit les mêmes privilèges qu'avant : la combinaison de tous les groupes auxquels leur ID appartient. Lorsque de nouvelles commandes **setmqaut** sont émises pour les ID utilisateur, elles sont immédiatement effectives.

Si vous créez un gestionnaire de files d'attente avec la règle utilisateur, ce gestionnaire de files d'attente dispose des droits uniquement pour l'utilisateur qui l'a créé (qui est normalement, mais pas nécessairement, l'ID utilisateur `mqm`). Il existe également des droits qui sont automatiquement accordés au groupe `mqm` . Toutefois, si vous ne disposez pas de `mqm` comme groupe principal, le groupe `mqm` n'est pas inclus dans l'ensemble d'autorisations initial.

Si vous passez d'une règle d'utilisateur à une règle de groupe, les droits basés sur l'utilisateur ne sont pas automatiquement supprimés. Cependant, ils ne sont plus utilisés lors de la vérification des droits. Avant de modifier les règles, enregistrez la configuration en cours, modifiez les règles, redémarrez le gestionnaire de files d'attente, puis réexécutez le script. Etant donné qu'il s'agit maintenant d'un gestionnaire de files d'attente basé sur le groupe, les règles de l'ID utilisateur sont stockées en fonction du groupe principal.

Concepts associés

Gestionnaire des droits d'accès aux objets (OAM)

Principaux et groupes sous UNIX, Linux et Windows

Section de service du fichier qm.ini

Référence associée

Commande **crtmqm** (créer le gestionnaire de files d'attente)

ULW Octroi de l'accès à un objet IBM MQ sur UNIX, Linux, and Windows

Utilisez la commande de contrôle **setmqaut** , la commande **SET AUTHREC** MQSC ou la commande PCF **MQCMD_SET_AUTH_REC** pour accorder aux utilisateurs et aux groupes d'utilisateurs l'accès aux objets IBM MQ . Notez que sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Pour une définition complète de la commande de contrôle **setmqaut** et de sa syntaxe, voir setmqaut.

Pour une définition complète de la commande **SET AUTHREC** MQSC et de sa syntaxe, voir SET AUTHREC.

Pour une définition complète de la commande PCF **MQCMD_SET_AUTH_REC** et de sa syntaxe, voir Définition de l'enregistrement des droits d'accès.

Le gestionnaire de files d'attente doit être en cours d'exécution pour pouvoir utiliser cette commande. Lorsque vous avez modifié l'accès à un principal, les modifications sont reflétées immédiatement par la méthode d'accès aux objets (OAM).

Pour accorder aux utilisateurs l'accès à un objet, vous devez spécifier:

- Nom du gestionnaire de files d'attente qui possède les objets que vous utilisez ; si vous ne spécifiez pas le nom d'un gestionnaire de files d'attente, le gestionnaire de files d'attente par défaut est utilisé.
- Nom et type de l'objet (pour identifier l'objet de manière unique). Vous spécifiez le nom en tant que *profil* ; Il s'agit soit du nom explicite de l'objet, soit d'un nom générique, incluant des caractères génériques. Pour une description détaillée des profils génériques et de l'utilisation des caractères génériques qu'ils contiennent, voir «Utilisation des profils génériques OAM sous UNIX, Linux, and Windows», à la page 365.
- Un ou plusieurs principaux et noms de groupe auxquels les droits s'appliquent.

  Si un ID utilisateur contient des espaces, placez-le entre guillemets lorsque vous utilisez cette commande. Sur les systèmes Windows , vous pouvez qualifier un ID utilisateur avec un nom de domaine. Si l'ID utilisateur réel contient un symbole arobase (@), remplacez-le par @ @ pour indiquer qu'il fait partie de l'ID utilisateur et non du délimiteur entre l'ID utilisateur et le nom de domaine.
- Liste des autorisations. Chaque élément de la liste indique un type d'accès qui doit être accordé à cet objet (ou révoqué). Chaque autorisation de la liste est indiquée en tant que mot clé, précédé d'un signe plus (+) ou d'un signe moins (-). Utilisez un signe plus pour ajouter l'autorisation spécifiée et un signe moins pour supprimer l'autorisation. Il ne doit pas y avoir d'espaces entre le signe + ou-et le mot clé.

  Vous pouvez spécifier n'importe quel nombre d'autorisations dans une seule commande. Par exemple, la liste des autorisations permettant à un utilisateur ou à un groupe de placer des messages dans une file d'attente et de les parcourir, mais de révoquer l'accès pour obtenir des messages est la suivante:

```
+browse -get +put
```

Exemples d'utilisation de la commande `setmqaut`

Les exemples suivants montrent comment utiliser la commande `setmqaut` pour accorder et révoquer des droits d'utilisation d'un objet:

```
setmqaut -m saturn.queue.manager -t queue -n RED.LOCAL.QUEUE
         -g groupa +browse -get +put
```

Dans cet exemple :

- `saturn.queue.manager` est le nom du gestionnaire de files d'attente
- `queue` est le type d'objet
- `RED.LOCAL.QUEUE` est le nom de l'objet
- `groupa` est l'identificateur du groupe avec les autorisations à modifier
- `+browse -get +put` est la liste d'autorisation pour la file d'attente spécifiée
  - `+browse` ajoute l'autorisation de parcourir les messages dans la file d'attente (pour émettre **MQGET** avec l'option de navigation)
  - `-get` supprime l'autorisation d'obtenir (**MQGET**) des messages de la file d'attente
  - `+put` ajoute l'autorisation d'insertion de messages (**MQPUT**) dans la file d'attente

La commande suivante révoque le droit d'insertion sur la file d'attente MyQueue du principal fvuser et des groupes groupa et groupb. Sur les systèmes UNIX and Linux , cette commande révoque également le droit d'insertion pour tous les principaux du même groupe principal que fvuser.

```
setmqaut -m saturn.queue.manager -t queue -n MyQueue -p fvuser
         -g groupa -g groupb -put
```

Utilisation de la commande setmqaut avec un service d'autorisation différent

Si vous utilisez votre propre service d'autorisation à la place de la méthode d'accès aux objets (OAM), vous pouvez spécifier le nom de ce service dans la commande **setmqaut** pour diriger la commande vers ce service. Vous devez spécifier ce paramètre si plusieurs composants installables sont en cours d'exécution en même temps ; si ce n'est pas le cas, la mise à jour est effectuée sur le premier composant installable pour le service d'autorisation. Par défaut, il s'agit de la méthode d'accès aux objets (OAM) fournie.

Remarques sur l'utilisation de la commande SET AUTHREC

La liste des autorisations à ajouter et la liste des autorisations à supprimer ne doivent pas se chevaucher. Par exemple, vous ne pouvez pas ajouter et supprimer des droits d'affichage avec la même commande. Cette règle s'applique même si les droits sont spécifiés à l'aide d'options différentes. Par exemple, la commande suivante échoue car le droit DSP et le droit ALLADM se chevauchent :

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALLADM)
```

Il existe une exception à ce comportement de chevauchement dans le cas du droit ALL. La commande suivante ajoute d'abord tous les droits ALL, puis supprime le droit SETID :

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(ALL) AUTHRMV(SETID)
```

La commande suivante supprime d'abord tous les droits ALL, puis ajoute le droit DSP :

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALL)
```

Quel que soit l'ordre dans lequel les droits sont indiqués dans la commande, les droits ALL sont traités en premier.

ULW Utilisation des profils génériques OAM sous UNIX, Linux, and Windows

Utilisez les profils génériques OAM pour définir, en une seule opération, les privilèges d'un utilisateur pour de nombreux objets, plutôt que d'avoir à émettre des commandes **setmqaut** ou **SET AUTHREC** distinctes pour chaque objet individuel lors de sa création. Notez que sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

L'utilisation de profils génériques dans les commandes setmqaut ou SET AUTHREC vous permet de définir des droits génériques pour tous les objets qui correspondent à ce profil.

Cette collection de rubriques décrit plus en détail l'utilisation des profils génériques.

Utilisation de caractères génériques dans les profils OAM

Ce qui rend un profil générique, c'est l'utilisation de caractères spéciaux (caractères génériques) dans le nom du profil. Par exemple, le caractère générique point d'interrogation (?) correspond à n'importe quel caractère unique dans un nom. Par conséquent, si vous spécifiez ABC.?EF, l'autorisation que vous accordez à ce profil s'applique à tous les objets portant les noms ABC.DEF, ABC.CEF, ABC.BEF, etc.

Les caractères génériques disponibles sont les suivants:

?

Utilisez le point d'interrogation (?) au lieu de n'importe quel caractère. Par exemple, `AB.?D` s'applique aux objets `AB.CD`, `AB.ED`et `AB.FD`.

*****

Utilisez l'astérisque (*) comme suit:

- Un *qualificateur* dans un nom de profil pour correspondre à un qualificateur dans un nom d'objet. Le qualificatif est une partie de nom d'objet délimitée par un point. Par exemple, dans `ABC.DEF.GHI`, les qualificatifs sont `ABC`, `DEF` et `GHI`.

  Par exemple, `ABC.*.JKL` s'applique aux objets `ABC.DEF.JKL`et `ABC.GHI.JKL`. (Notez qu'il ne s'applique **pas** à `ABC.JKL` ; * utilisé dans ce contexte indique toujours un qualificateur.)
- Caractère dans un qualificatif d'un nom de profil qui doit correspondre à zéro ou plusieurs caractères dans le qualificatif d'un nom d'objet.

  Par exemple, `ABC.DE*.JKL` s'applique aux objets `ABC.DE.JKL`, `ABC.DEF.JKL`et `ABC.DEGH.JKL`.

******

Utilisez le double astérisque (**) **une fois** dans un nom de profil comme suit:

- Nom de profil complet correspondant à tous les noms d'objet. Par exemple, si vous utilisez `-t prcs` pour identifier les processus, puis utilisez ** comme nom de profil, vous modifiez les autorisations pour tous les processus.
- Comme qualificatif de début, de milieu ou de fin dans un nom de profil pour correspondre à zéro ou plusieurs qualificatifs dans un nom d'objet. Par exemple, `**.ABC` identifie tous les objets avec le qualificateur final ABC.

Vous ne pouvez utiliser que le double astérisque ** comme qualificateur complet:

```
**.DEF
  ABC.**
  A*.**
```

mais pas en tant que

```
A**
```

sinon, vous recevez le message AMQ7226E: `Le nom de profil n'est pas valide.`

Remarque : Lorsque vous utilisez des caractères génériques sur les systèmes UNIX et Linux , vous **devez** placer le nom de profil entre apostrophes.

Priorités de profil

Un point important à comprendre lors de l'utilisation de profils génériques est la priorité donnée aux profils lors de la détermination des droits à appliquer à un objet en cours de création. Par exemple, supposons que vous ayez émis les commandes suivantes:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

Le premier accorde le droit d'insertion à toutes les files d'attente pour le principal fred dont les noms correspondent au profil AB. * ; La seconde permet d'obtenir des droits sur les mêmes types de file d'attente qui correspondent au profil AB.C*.

Supposons que vous créiez maintenant une file d'attente appelée AB.CD. Selon les règles de correspondance des caractères génériques, setmqaut peut s'appliquer à cette file d'attente. Alors, a-t-elle mis ou obtenu l'autorité?

Pour trouver la réponse, vous appliquez la règle selon laquelle, chaque fois que plusieurs profils peuvent s'appliquer à un objet, **seuls les plus spécifiques s'appliquent**. La façon dont vous appliquez cette règle consiste à comparer les noms de profil de gauche à droite. Lorsqu'ils diffèrent, un caractère non

générique est plus spécifique qu'un caractère générique. Ainsi, dans cet exemple, la file d'attente AB.CD dispose du droit **get** (AB.C* est plus spécifique que AB. *).

Lorsque vous comparez des caractères génériques, l'ordre de la *spécificité* est le suivant:

1. ?
2. *
3. **

Vidage des paramètres de profil

Pour une définition complète de la commande de contrôle **dmpmqaut** et de sa syntaxe, voir dmpmqaut.

Pour une définition complète de la commande **DISPLAY AUTHREC** MQSC et de sa syntaxe, voir DISPLAY AUTHREC.

Pour une définition complète de la commande PCF **MQCMD_INQUIRE_AUTH_RECS** et de sa syntaxe, voir Inquire Authority Records.

Les exemples suivants illustrent l'utilisation de la commande de contrôle **dmpmqaut** pour vider des enregistrements de droits d'accès pour des profils génériques:

1. Cet exemple vide tous les enregistrements de droits d'accès dont le profil correspond à la file d'attente a.b.c pour le principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   Le vidage résultant se présente comme suit:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Remarque : Bien que les utilisateurs sous UNIX et Linux puissent utiliser l'option `-p` pour la commande **dmpmqaut** , ils doivent utiliser `-g groupname` à la place lors de la définition des autorisations.
2. Cet exemple vide tous les enregistrements de droits d'accès dont le profil correspond à la file d'attente a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   Le vidage résultant se présente comme suit:

   ```
   profile:     a.b.c
   object type: queue
   entity:      Administrator
   type:        principal
   authority:   all
   - - - - - - - - - - - - - - - -
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   - - - - - - - - - - - - - - - -
   profile:     a.**
   object type: queue
   entity:      group1
   type:        group
   authority:   get
   ```
3. Cet exemple vide tous les enregistrements de droits d'accès pour le profil a.b. *, de type file d'attente.

```
dmpmqaut -m qmgr1 -n a.b.* -t q
```

Le vidage résultant se présente comme suit:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

4. Cet exemple vide tous les enregistrements de droits d'accès pour le gestionnaire de files d'attente qmX.

```
dmpmqaut -m qmX
```

Le vidage résultant se présente comme suit:

```
profile:     q1
object type: queue
entity:      Administrator
type:        principal
authority:   all
- - - - - - - - - - - - - - - -
profile:     q*
object type: queue
entity:      user1
type:        principal
authority:   get, browse
- - - - - - - - - - - - - - - -
profile:     name.*
object type: namelist
entity:      user2
type:        principal
authority:   get
- - - - - - - - - - - - - - - -
profile:     pr1
object type: process
entity:      group1
type:        group
authority:   get
```

5. Cet exemple vide tous les noms de profil et tous les types d'objet pour le gestionnaire de files d'attente qmX.

```
dmpmqaut -m qmX -l
```

Le vidage résultant se présente comme suit:

```
profile: q1, type: queue
profile: q*, type: queue
profile: name.*, type: namelist
profile: pr1, type: process
```

Remarque : Pour IBM MQ for Windows uniquement, tous les principaux affichés incluent des informations de domaine, par exemple:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

ULW *Utilisation de caractères génériques dans les profils OAM sous UNIX, Linux, and Windows*

Utilisez des caractères génériques dans un nom de profil de gestionnaire des droits d'accès aux objets (OAM) pour rendre ce profil applicable à plusieurs objets.

Ce qui rend un profil générique, c'est l'utilisation de caractères spéciaux (caractères génériques) dans le nom du profil. Par exemple, le caractère générique point d'interrogation (?) correspond à n'importe quel caractère unique dans un nom. Par conséquent, si vous spécifiez `ABC.?EF`, l'autorisation que vous accordez à ce profil s'applique à tous les objets portant les noms `ABC.DEF`, `ABC.CEF`, `ABC.BEF`, etc.

Les caractères génériques disponibles sont les suivants:

?

Utilisez le point d'interrogation (?) au lieu de n'importe quel caractère. Par exemple, `AB.?D` s'applique aux objets `AB.CD`, `AB.EDet AB.FD`.

*****

Utilisez l'astérisque (*) comme suit:

- Un *qualificateur* dans un nom de profil pour correspondre à un qualificateur dans un nom d'objet. Le qualificatif est une partie de nom d'objet délimitée par un point. Par exemple, dans `ABC.DEF.GHI`, les qualificatifs sont `ABC`, `DEF` et `GHI`.

  Par exemple, `ABC.*.JKL` s'applique aux objets `ABC.DEF.JKLet ABC.GHI.JKL`. (Notez qu'il ne s'applique **pas** à `ABC.JKL` ; * utilisé dans ce contexte indique toujours un qualificateur.)
- Caractère dans un qualificatif d'un nom de profil qui doit correspondre à zéro ou plusieurs caractères dans le qualificatif d'un nom d'objet.

  Par exemple, `ABC.DE*.JKL` s'applique aux objets `ABC.DE.JKL`, `ABC.DEF.JKLet ABC.DEGH.JKL`.

******

Utilisez le double astérisque (**) **une fois** dans un nom de profil comme suit:

- Nom de profil complet correspondant à tous les noms d'objet. Par exemple, si vous utilisez `-t prcs` pour identifier les processus, puis utilisez ** comme nom de profil, vous modifiez les autorisations pour tous les processus.
- Comme qualificatif de début, de milieu ou de fin dans un nom de profil pour correspondre à zéro ou plusieurs qualificatifs dans un nom d'objet. Par exemple, `**.ABC` identifie tous les objets avec le qualificateur final ABC.

Remarque : Lorsque vous utilisez des caractères génériques sur des systèmes UNIX and Linux , vous **devez** placer le nom de profil entre apostrophes.

ULW *Priorités de profil sous UNIX, Linux, and Windows*

Plusieurs profils génériques peuvent s'appliquer à un seul objet. Lorsque c'est le cas, la règle la plus spécifique s'applique.

Un point important à comprendre lors de l'utilisation de profils génériques est la priorité donnée aux profils lors de la détermination des droits à appliquer à un objet en cours de création. Par exemple, supposons que vous ayez émis les commandes suivantes:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

Le premier accorde le droit d'insertion à toutes les files d'attente pour le principal fred dont les noms correspondent au profil AB. * ; La seconde permet d'obtenir des droits sur les mêmes types de file d'attente qui correspondent au profil AB.C*.

Supposons que vous créiez maintenant une file d'attente appelée AB.CD. Selon les règles de correspondance des caractères génériques, setmqaut peut s'appliquer à cette file d'attente. Alors, a-t-elle mis ou obtenu l'autorité?

Pour trouver la réponse, vous appliquez la règle selon laquelle, chaque fois que plusieurs profils peuvent s'appliquer à un objet, **seuls les plus spécifiques s'appliquent**. La façon dont vous appliquez cette règle consiste à comparer les noms de profil de gauche à droite. Lorsqu'ils diffèrent, un caractère non générique est plus spécifique qu'un caractère générique. Ainsi, dans cet exemple, la file d'attente AB.CD dispose du droit **get** (AB.C* est plus spécifique que AB. *).

Lorsque vous comparez des caractères génériques, l'ordre de la *spécificité* est le suivant:

1. ?
2. *
3. **

Pour obtenir des informations équivalentes lors de l'utilisation de cette commande MQSC, voir SET AUTHREC .

ULW *Vidage des paramètres de profil sous UNIX, Linux, and Windows*

Utilisez la commande de contrôle **dmpmqaut** , la commande **DISPLAY AUTHREC** MQSC ou la commande **MQCMD_INQUIRE_AUTH_RECS** PCF pour vider les autorisations en cours associées à un profil spécifié. Notez que sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **DISPLAY AUTHREC** .

Pour une définition complète de la commande de contrôle **dmpmqaut** et de sa syntaxe, voir dmpmqaut.

Pour une définition complète de la commande **DISPLAY AUTHREC** MQSC et de sa syntaxe, voir DISPLAY AUTHREC.

Pour une définition complète de la commande PCF **MQCMD_INQUIRE_AUTH_RECS** et de sa syntaxe, voir Inquire Authority Records.

Les exemples suivants illustrent l'utilisation de la commande de contrôle **dmpmqaut** pour vider des enregistrements de droits d'accès pour des profils génériques:

1. Cet exemple vide tous les enregistrements de droits d'accès dont le profil correspond à la file d'attente a.b.c pour le principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   Le vidage résultant ressemble à l'exemple suivant:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Remarque : Les utilisateurs UNIX and Linux ne peuvent pas utiliser l'option `-p` ; ils doivent utiliser `-g groupname` à la place.
2. Cet exemple vide tous les enregistrements de droits d'accès dont le profil correspond à la file d'attente a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   Le vidage résultant ressemble à l'exemple suivant:

   ```
   profile:     a.b.c
   object type: queue
   entity:      Administrator
   type:        principal
   authority:   all
   - - - - - - - - - - - - - - - - -
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   ```

```
authority:   get, browse, put, inq
- - - - - - - - - - - - - - - - -
profile:     a.**
object type: queue
entity:      group1
type:        group
authority:   get
```

3. Cet exemple vide tous les enregistrements de droits d'accès pour le profil a.b. *, de type file d'attente.

```
dmpmqaut -m qmgr1 -n a.b.* -t q
```

Le vidage résultant ressemble à l'exemple suivant:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

4. Cet exemple vide tous les enregistrements de droits d'accès pour le gestionnaire de files d'attente qmX.

```
dmpmqaut -m qmX
```

Le vidage résultant ressemble à l'exemple suivant:

```
profile:     q1
object type: queue
entity:      Administrator
type:        principal
authority:   all
- - - - - - - - - - - - - - - - -
profile:     q*
object type: queue
entity:      user1
type:        principal
authority:   get, browse
- - - - - - - - - - - - - - - - -
profile:     name.*
object type: namelist
entity:      user2
type:        principal
authority:   get
- - - - - - - - - - - - - - - - -
profile:     pr1
object type: process
entity:      group1
type:        group
authority:   get
```

5. Cet exemple vide tous les noms de profil et tous les types d'objet pour le gestionnaire de files d'attente qmX.

```
dmpmqaut -m qmX -l
```

Le vidage résultant ressemble à l'exemple suivant:

```
profile: q1, type: queue
profile: q*, type: queue
profile: name.*, type: namelist
profile: pr1, type: process
```

Remarque : Pour IBM MQ for Windows uniquement, tous les principaux affichés incluent des informations de domaine, par exemple:

```
profile:     a.b.*
```

```
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

ULW Affichage des paramètres d'accès sur UNIX, Linux, and Windows

Utilisez la commande de contrôle **dspmqaut** , la commande **DISPLAY AUTHREC** MQSC ou la commande **MQCMD_INQUIRE_ENTITY_AUTH** PCF pour afficher les autorisations dont dispose un principal ou un groupe spécifique pour un objet particulier. Notez que sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **DISPLAY AUTHREC** .

Le gestionnaire de files d'attente doit être en cours d'exécution pour pouvoir utiliser cette commande. Lorsque vous modifiez l'accès à un principal, les modifications sont répercutées immédiatement par la méthode d'accès aux objets (OAM). L'autorisation ne peut être affichée que pour un seul groupe ou principal à la fois.

Pour une définition complète de la commande de contrôle **dmpmqaut** et de sa syntaxe, voir dmpmqaut.

Pour une définition complète de la commande **DISPLAY AUTHREC** MQSC et de sa syntaxe, voir DISPLAY AUTHREC.

Pour une définition complète de la commande PCF **MQCMD_INQUIRE_AUTH_RECS** et de sa syntaxe, voir Inquire Authority Records.

L'exemple suivant illustre l'utilisation de la commande de contrôle **dspmqaut** pour afficher les autorisations dont dispose le groupe `GpAdmin` pour une définition de processus nommée `Annuities` qui se trouve sur le gestionnaire de files d'attente `QueueMan1`.

```
dspmqaut -m QueueMan1 -t process -n Annuities -g GpAdmin
```

ULW Modification et révocation de l'accès à un objet IBM MQ sous UNIX, Linux, and Windows

Pour modifier le niveau d'accès d'un utilisateur ou d'un groupe à un objet, utilisez la commande de contrôle **setmqaut** , la commande **DELETE AUTHREC** MQSC ou la commande PCF **MQCMD_DELETE_AUTH_REC** . MQ Appliance Notez que sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **DELETE AUTHREC** .

Le processus de suppression de l'utilisateur d'un groupe est décrit dans:

- Windows «Création et gestion de groupes sur Windows», à la page 147
- AIX «Création et gestion de groupes sur AIX», à la page 145
- Solaris «Création et gestion de groupes sur Solaris», à la page 147
- Linux «Création et gestion de groupes sur Linux», à la page 146

.

L'ID utilisateur qui crée un objet IBM MQ dispose de droits de contrôle complets sur cet objet. Si vous supprimez cet ID utilisateur du groupe mqm local (ou du groupe Administrateurs sur les systèmes Windows), ces droits ne sont pas révoqués. Utilisez la commande de contrôle **setmqaut** ou la commande PCF **MQCMD_DELETE_AUTH_REC** pour révoquer l'accès à un objet pour l'ID utilisateur qui l'a créé, après l'avoir supprimé du groupe mqm ou Administrateurs.

Pour une définition complète de la commande de contrôle setmqaut et de sa syntaxe, voir setmqaut.

Pour une définition complète de la commande **DELETE AUTHREC** MQSC et de sa syntaxe, voir DELETE AUTHREC.

Pour une définition complète de la commande PCF **MQCMD_DELETE_AUTH_REC** et de sa syntaxe, voir Delete Authority Record.

Windows Sous Windows, depuis IBM MQ 8.0, vous pouvez supprimer à tout moment les entrées OAM correspondant à un compte utilisateur Windows particulier à l'aide du paramètre **-u** *SID* de **setmqaut**.

Avant IBM MQ 8.0, vous deviez supprimer les entrées OAM correspondant à un compte utilisateur Windows particulier avant de supprimer le profil utilisateur. Il était impossible de supprimer les entrées OAM après la suppression du compte utilisateur.

ULW Prévention des contrôles d'accès de sécurité sur les systèmes UNIX, Linux, and Windows

Pour désactiver toutes les vérifications de sécurité, vous pouvez désactiver le gestionnaire des droits d'accès aux objets (OAM). Cela peut convenir à un environnement de test. Après avoir désactivé ou supprimé la méthode d'accès aux objets (OAM), vous ne pouvez pas ajouter une méthode d'accès aux objets (OAM) à un gestionnaire de files d'attente existant.

Si vous décidez de ne pas effectuer de contrôles de sécurité (par exemple, dans un environnement de test), vous pouvez désactiver la méthode d'accès aux objets (OAM) de l'une des deux manières suivantes:

- Avant de créer un gestionnaire de files d'attente, définissez la variable d'environnement du système d'exploitation MQSNOAUT.

  Pour plus d'informations sur les implications de la définition de la variable MQSNOAUT et sur la manière de définir MQSNOAUT sur Windows et UNIX, voir Description des variables d'environnement .
- Editez le fichier de configuration du gestionnaire de files d'attente pour supprimer le service.

Si vous utilisez la commande **setmqaut**ou **dspmqaut** alors que la méthode d'accès aux objets (OAM) est désactivée, notez les points suivants:

- La méthode d'accès aux objets (OAM) ne valide pas le principal ou le groupe spécifié, ce qui signifie que la commande peut accepter des valeurs non valides.
- La méthode d'accès aux objets (OAM) n'effectue pas de contrôles de sécurité et indique que tous les principaux et groupes sont autorisés à effectuer toutes les opérations d'objet applicables.

Avertissement : Lorsqu'une méthode d'accès aux objets (OAM) est supprimée, elle ne peut pas être remise sur un gestionnaire de files d'attente existant. En effet, la méthode d'accès aux objets (OAM) doit être en place au moment de la création de l'objet. Pour utiliser à nouveau la méthode d'accès aux objets (OAM) IBM MQ une fois qu'elle a été supprimée, régénérez le gestionnaire de files d'attente.

Concepts associés

Services et composants installables pour UNIX, Linux et Windows

Tâches associées

Configuration des services installables

Référence associée

Informations de référence sur les services installables

Octroi de l'accès requis aux ressources

Cette rubrique permet de déterminer les tâches à effectuer pour appliquer la sécurité à votre système IBM MQ sous UNIX, Linux, Windows, IBM iet z/OS.

Pourquoi et quand exécuter cette tâche

Au cours de cette tâche, vous décidez des actions nécessaires pour appliquer le niveau de sécurité approprié aux éléments de votre installation IBM MQ . Chaque tâche individuelle à qui vous vous référez fournit des instructions étape par étape pour toutes les plateformes.

Procédure

1. Avez-vous besoin de limiter l'accès à votre gestionnaire de files d'attente à certains utilisateurs?
   a) Non: ne prenez aucune autre mesure.
   b) Oui: Passez à la question suivante.
2. Ces utilisateurs ont-ils besoin d'un accès administratif partiel sur un sous-ensemble de ressources de gestionnaire de files d'attente?
   a) Non: Passez à la question suivante.
   b) Oui: voir «Octroi d'un accès administrateur partiel sur un sous-ensemble de ressources de gestionnaire de files d'attente», à la page 374.
3. Ces utilisateurs ont-ils besoin d'un accès administrateur complet sur un sous-ensemble de ressources de gestionnaire de files d'attente?
   a) Non: Passez à la question suivante.
   b) Oui: voir «Octroi d'un accès administrateur complet sur un sous-ensemble de ressources de gestionnaire de files d'attente», à la page 384.
4. Ces utilisateurs ont-ils besoin d'un accès en lecture seule à toutes les ressources du gestionnaire de files d'attente?
   a) Non: Passez à la question suivante.
   b) Oui: voir «Octroi d'un accès en lecture seule à toutes les ressources d'un gestionnaire de files d'attente», à la page 392.
5. Ces utilisateurs ont-ils besoin d'un accès administrateur complet sur toutes les ressources du gestionnaire de files d'attente?
   a) Non: Passez à la question suivante.
   b) Oui: voir «Octroi d'un accès administrateur complet à toutes les ressources d'un gestionnaire de files d'attente», à la page 393.
6. Avez-vous besoin d'applications utilisateur pour vous connecter à votre gestionnaire de files d'attente?
   a) Non: Désactiver la connectivité, comme décrit dans «Suppression de la connectivité au gestionnaire de files d'attente», à la page 395
   b) Oui: voir «Autorisation des applications utilisateur à se connecter à votre gestionnaire de files d'attente», à la page 396.

Multi z/OS Octroi d'un accès administrateur partiel sur un sous-ensemble de ressources de gestionnaire de files d'attente

Vous devez accorder à certains utilisateurs un accès administrateur partiel à certaines ressources de gestionnaire de files d'attente, mais pas à toutes. Utilisez ce tableau pour déterminer les actions que vous devez effectuer.

Tableau 69. Octroi d'un accès administrateur partiel à un sous-ensemble de ressources de gestionnaire de files d'attente

Les utilisateurs doivent administrer les objets de ce type	**Effectuer cette action**
Files d'attente	Accordez un accès administrateur partiel aux files d'attente requises, comme décrit dans «Octroi d'un accès administrateur limité à certaines files d'attente», à la page 375
Rubriques	Accordez un accès administrateur partiel aux rubriques requises, comme décrit dans «Octroi d'un accès administrateur limité à certaines rubriques», à la page 377

Tableau 69. Octroi d'un accès administrateur partiel à un sous-ensemble de ressources de gestionnaire de files d'attente (suite)

Les utilisateurs doivent administrer les objets de ce type	**Effectuer cette action**
Canaux	Accordez un accès administrateur partiel aux canaux requis, comme décrit dans «Octroi d'un accès administratif limité à certains canaux», à la page 378
Gestionnaire de files d'attente	Accordez un accès administrateur partiel au gestionnaire de files d'attente, comme décrit dans «Octroi d'un accès administrateur limité à un gestionnaire de files d'attente», à la page 379
Processus	Accordez un accès administratif partiel aux processus requis, comme décrit dans «Octroi d'un accès administrateur limité à certains processus», à la page 380
Listes de noms	Accordez un accès administrateur partiel aux listes de noms requises, comme décrit dans «Octroi d'un accès administrateur limité à certaines listes de noms», à la page 382
Services	Accordez un accès administrateur partiel aux services requis, comme décrit dans «Octroi d'un accès administratif limité à certains services», à la page 383

Octroi d'un accès administrateur limité à certaines files d'attente

Accordez un accès administrateur partiel à certaines files d'attente d'un gestionnaire de files d'attente, à chaque groupe d'utilisateurs qui en ont besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certaines files d'attente pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW

  Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName ReqdAction
```

- IBM i
  Pour IBM i, exécutez la commande suivante:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS Pour z/OS, exécutez les commandes suivantes pour accorder l'accès à une file d'attente spécifiée:

  ```
  RDEFINE MQADMIN QMgrName.QUEUE. ObjectProfile UACC(NONE)
  PERMIT QMgrName.QUEUE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Pour spécifier les commandes MQSC que l'utilisateur peut exécuter sur la file d'attente, émettez les commandes suivantes pour chaque commande MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction. QType UACC(NONE)
  PERMIT QMgrName. ReqdAction. QType CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Pour permettre à l'utilisateur d'utiliser la commande DISPLAY QUEUE, exécutez les commandes suivantes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY. QType UACC(NONE)
  PERMIT QMgrName.DISPLAY. QType CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

  ReqdAction
  L'action que vous autorisez le groupe à effectuer:

  - ULW Sur les systèmes UNIX, Linux, and Windows , toute combinaison des autorisations suivantes: + chg, + clr, + dlt, + dsp. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
  - IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMDLT, *ADMDSP. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.
  - z/OS Sous z/OS, l'une des valeurs ALTER, CLEAR, DELETE ou MOVE.

  Remarque : L'octroi + crt pour les files d'attente fait indirectement de l'utilisateur ou du groupe un administrateur. N'utilisez pas le droit + crt pour accorder un accès administrateur limité à certaines files d'attente.

  QType

  Pour la commande DISPLAY, l'une des valeurs QUEUE, QLOCAL, QALIAS, QMODEL, QREMOTE ou QCLUSTER.

  Pour les autres valeurs de *ReqdAction*, l'une des valeurs QLOCAL, QALIAS, QMODEL ou QREMOTE.

Octroi d'un accès administrateur limité à certaines rubriques

Accordez un accès administratif partiel à certaines rubriques d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certaines rubriques pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW

  Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName ReqdAction
  ```

- IBM i

  Pour IBM i, exécutez la commande suivante:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS Pour z/OS, exécutez les commandes suivantes:

  ```
  RDEFINE MQADMIN QMgrName.TOPIC. ObjectProfile UACC(NONE)
  PERMIT QMgrName.TOPIC. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Ces commandes permettent d'accéder à la rubrique spécifiée. Pour déterminer les commandes MQSC que l'utilisateur peut exécuter sur la rubrique, exécutez les commandes suivantes pour chaque commande MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.TOPIC UACC(NONE)
  PERMIT QMgrName. ReqdAction.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Pour permettre à l'utilisateur d'utiliser la commande DISPLAY TOPIC, exécutez les commandes suivantes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.TOPIC UACC(NONE)
  PERMIT QMgrName.DISPLAY.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

ReqdAction
L'action que vous autorisez le groupe à effectuer:

- ULW Sur les systèmes UNIX, Linux, and Windows , toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + dsp. + ctrl. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
- IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.
- z/OS Sous z/OS, l'une des valeurs ALTER, CLEAR, DEFINE, DELETE ou MOVE.

Octroi d'un accès administratif limité à certains canaux

Accordez un accès administratif partiel à certains canaux d'un gestionnaire de files d'attente, à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certains canaux pour certaines actions, utilisez les commandes appropriées à votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName ReqdAction
```

- IBM i
  Sous IBM i :

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*CHL) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS Sous z/OS :

```
RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
PERMIT QMgrName.CHANNEL. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Ces commandes permettent d'accéder au canal spécifié. Pour déterminer les commandes MQSC que l'utilisateur peut exécuter sur le canal, exécutez les commandes suivantes pour chaque commande MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.CHANNEL UACC(NONE)
PERMIT QMgrName. ReqdAction.CHANNEL CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Pour permettre à l'utilisateur d'utiliser la commande DISPLAY CHANNEL, exécutez les commandes suivantes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.CHANNEL UACC(NONE)
PERMIT QMgrName.DISPLAY.CHANNEL CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

ReqdAction
L'action que vous autorisez le groupe à effectuer:

- ULW Sous UNIX, Linux, and Windows, toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + dsp. + ctrl, + ctrlx. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
- IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLx. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.
- z/OS Sous z/OS, l'une des valeurs ALTER, CLEAR, DEFINE, DELETE ou MOVE.

Octroi d'un accès administrateur limité à un gestionnaire de files d'attente

Accordez un accès administrateur partiel à un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité afin d'effectuer certaines actions sur le gestionnaire de files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW

  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName ReqdAction
  ```

- IBM i

  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS

  Sous z/OS :

  Pour déterminer les commandes MQSC que vous pouvez exécuter sur le gestionnaire de files d'attente, exécutez les commandes suivantes pour chaque commande MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.QMGR UACC(NONE)
  PERMIT QMgrName. ReqdAction.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Pour permettre à l'utilisateur d'utiliser la commande DISPLAY QMGR, exécutez les commandes suivantes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.QMGR UACC(NONE)
  PERMIT QMgrName.DISPLAY.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

  ReqdAction
  L'action que vous autorisez le groupe à effectuer:

  – ULW Sous UNIX, Linux, and Windows, toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + dsp. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.

    Bien que + set soit une autorisation MQI et qu'elle ne soit normalement pas considérée comme administrative, l'octroi de + set sur le gestionnaire de files d'attente peut indirectement conduire à des droits d'administration complets. N'accordez pas + défini aux utilisateurs et aux applications ordinaires.

  – IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.

Octroi d'un accès administrateur limité à certains processus

Accordez un accès administratif partiel à certains processus d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certains processus pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName ReqdAction
  ```

- IBM i
  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS Sous z/OS :

  ```
  RDEFINE MQADMIN QMgrName.PROCESS. ObjectProfile UACC(NONE)
  PERMIT QMgrName.PROCESS. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Ces commandes permettent d'accéder au canal spécifié. Pour déterminer les commandes MQSC que l'utilisateur peut exécuter sur le canal, exécutez les commandes suivantes pour chaque commande MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.PROCESS UACC(NONE)
  PERMIT QMgrName. ReqdAction.PROCESS CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Pour permettre à l'utilisateur d'utiliser la commande DISPLAY PROCESS, exécutez les commandes suivantes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.PROCESS UACC(NONE)
  PERMIT QMgrName.DISPLAY.PROCESS CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

ReqdAction
L'action que vous autorisez le groupe à effectuer:

- ULW Sous UNIX, Linux, and Windows, toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + dsp. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
- IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.
- z/OS Sous z/OS, l'une des valeurs ALTER, CLEAR, DEFINE, DELETE ou MOVE.

Octroi d'un accès administrateur limité à certaines listes de noms

Accordez un accès administrateur partiel à certaines listes de noms sur un gestionnaire de files d'attente, à chaque groupe d'utilisateurs qui en ont besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certaines listes de noms pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

```
setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName ReqdAction
```

- IBM i
  Sous IBM i :

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*NMLIST) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS Sous z/OS :

```
RDEFINE MQADMIN QMgrName.NAMELIST. ObjectProfile UACC(NONE)
PERMIT QMgrName.NAMELIST. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

  Ces commandes permettent d'accéder à la liste de noms spécifiée. Pour déterminer les commandes MQSC que l'utilisateur peut exécuter sur la liste de noms, exécutez les commandes suivantes pour chaque commande MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.NAMELIST UACC(NONE)
PERMIT QMgrName. ReqdAction.NAMELIST CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Pour permettre à l'utilisateur d'utiliser la commande DISPLAY NAMELIST, exécutez les commandes suivantes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.NAMELIST UACC(NONE)
PERMIT QMgrName.DISPLAY.NAMELIST CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

ReqdAction
L'action que vous autorisez le groupe à effectuer:

- ULW Sous UNIX, Linux, and Windows, toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
- IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLX. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.
- z/OS Sous z/OS, l'une des valeurs ALTER, CLEAR, DEFINE, DELETE ou MOVE.

Octroi d'un accès administratif limité à certains services

Accordez un accès administratif partiel à certains services d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur limité à certains services pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation. z/OS Notez que les objets de service n'existent pas sur z/OS.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName ReqdAction
  ```

- Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*SVC) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS Sous z/OS :

  Ces commandes permettent d'accéder au service spécifié. Pour déterminer les commandes MQSC que l'utilisateur peut exécuter sur le service, exécutez les commandes suivantes pour chaque commande MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.SERVICE UACC(NONE)
  PERMIT QMgrName. ReqdAction.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Pour permettre à l'utilisateur d'utiliser la commande DISPLAY SERVICE, exécutez les commandes suivantes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.SERVICE UACC(NONE)
  PERMIT QMgrName.DISPLAY.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

  ReqdAction
  L'action que vous autorisez le groupe à effectuer:

  - ULW Sur les systèmes UNIX, Linux, and Windows , toute combinaison des autorisations suivantes: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. L'autorisation + alladm est équivalente à + chg + clr + dlt + dsp.
  - IBM i Sous IBM i, toute combinaison des droits suivants: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLX. L'autorisation *ALLADM est équivalente à toutes ces autorisations individuelles.

Octroi d'un accès administrateur complet sur un sous-ensemble de ressources de gestionnaire de files d'attente

Vous devez accorder à certains utilisateurs un accès administrateur complet à certaines ressources de gestionnaire de files d'attente, mais pas à toutes. Utilisez ces tableaux pour déterminer les actions que vous devez effectuer.

Tableau 70. Octroi d'un accès administrateur complet à un sous-ensemble de ressources de gestionnaire de files d'attente

Les utilisateurs doivent administrer les objets de ce type	**Effectuer cette action**
Files d'attente	Accordez un accès administrateur complet aux files d'attente requises, comme décrit dans «Octroi d'un accès administrateur complet à certaines files d'attente», à la page 385
Rubriques	Accordez un accès administrateur complet aux rubriques requises, comme décrit dans «Octroi d'un accès administrateur complet à certaines rubriques», à la page 386
Canaux	Accordez un accès administrateur complet aux canaux requis, comme décrit dans «Octroi d'un accès administrateur complet à certains canaux», à la page 387
Gestionnaire de files d'attente	Accordez un accès administrateur complet au gestionnaire de files d'attente, comme décrit dans «Octroi d'un accès administrateur complet à un gestionnaire de files d'attente», à la page 388
Processus	Accordez un accès administrateur complet aux processus requis, comme décrit dans «Octroi d'un accès administrateur complet à certains processus», à la page 389
Listes de noms	Accordez un accès administrateur complet aux listes de noms requises, comme décrit dans «Octroi d'un accès administrateur complet à certaines listes de noms», à la page 390
Services	Accordez un accès administrateur complet aux services requis, comme décrit dans «Octroi d'un accès administrateur complet à certains services», à la page 391

Octroi d'un accès administrateur complet à certaines files d'attente

Accordez un accès administrateur complet à certaines files d'attente d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certaines files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i
- Linux
- UNIX
- Windows

Remarque : Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +alladm
```

- IBM i
  Sous IBM i :

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*ALLADM) MQMNAME(' QMgrName
')
```

- z/OS
  Sous z/OS :

```
RDEFINE MQADMIN QMgrName.QUEUE. ObjectProfile UACC(NONE)
PERMIT QMgrName.QUEUE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à certaines rubriques

Accordez un accès administrateur complet à certaines rubriques d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certaines rubriques pour certaines actions, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +alladm
```

- IBM i
  Sous IBM i :

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(ALLADM) MQMNAME('
QMgrName ')
```

- z/OS
  Sous z/OS :

```
RDEFINE MQADMIN QMgrName.TOPIC. ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à certains canaux

Accordez un accès administrateur complet à certains canaux d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certains canaux, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName +alladm
```

- IBM i
  Sous IBM i :

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*CHL) USER(GroupName) AUT(ALLADM) MQMNAME('
QMgrName ')
```

- z/OS

  Sous z/OS :

  ```
  RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
  PERMIT QMgrName.CHANNEL. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à un gestionnaire de files d'attente

Accordez un accès administrateur complet à un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet au gestionnaire de files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW

  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -t qmgr -g GroupName +alladm
  ```

- IBM i

  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS

Sous z/OS :

```
RDEFINE MQADMIN QMgrName.QMGR UACC(NONE)
PERMIT QMgrName.QMGR CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à certains processus

Accordez un accès administrateur complet à certains processus d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certains processus, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +alladm
  ```

- IBM i
  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS
  Sous z/OS :

  ```
  RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
  PERMIT QMgrName.PROCESS. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à certaines listes de noms

Accordez un accès administrateur complet à certaines listes de noms d'un gestionnaire de files d'attente, à chaque groupe d'utilisateurs qui en ont besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certaines listes de noms, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName +alladm
  ```

- IBM i
  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS
  Sous z/OS :

  ```
  RDEFINE MQADMIN QMgrName.NAMELIST. ObjectProfile UACC(NONE)
  PERMIT QMgrName.NAMELIST. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à certains services

Accordez un accès administrateur complet à certains services d'un gestionnaire de files d'attente à chaque groupe d'utilisateurs qui en ont besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder un accès administrateur complet à certains services, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName +alladm
  ```

- IBM i
  Sous IBM i :

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*SVC) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS
  Sous z/OS :

  ```
  RDEFINE MQADMIN QMgrName.SERVICE. ObjectProfile UACC(NONE)
  PERMIT QMgrName.SERVICE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès en lecture seule à toutes les ressources d'un gestionnaire de files d'attente

Accordez un accès en lecture seule à toutes les ressources d'un gestionnaire de files d'attente, à chaque utilisateur ou groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Utilisez l'assistant d'ajout de droits basés sur les rôles ou les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Après avoir modifié les détails d'autorisation, effectuez une actualisation de la sécurité à l'aide de la commande REFRESH SECURITY .

Procédure

- A l'aide de l'assistant:
  a) Dans le panneau IBM MQ Explorer Navigator , cliquez avec le bouton droit de la souris sur le gestionnaire de files d'attente, puis cliquez sur **Droits sur les objets** > **Ajouter des droits basés sur les rôles**

  L'assistant Ajout de droits basés sur des rôles s'ouvre.
- Windows UNIX
  Pour les systèmes UNIX et Windows , exécutez les commandes suivantes:

```
setmqaut -m QMgrName -n ** -t queue -g GroupName +browse +dsp
setmqaut -m QMgrName -n SYSTEM.ADMIN.COMMAND.QUEUE -t queue -g GroupName +dsp +inq +put
setmqaut -m QMgrName -n SYSTEM.MQEXPLORER.REPLY.MODEL -t queue -g GroupName +dsp +inq +get
+put
setmqaut -m QMgrName -n ** -t topic -g GroupName +dsp
setmqaut -m QMgrName -n ** -t channel -g GroupName +dsp +inq
setmqaut -m QMgrName -n ** -t clntconn -g GroupName +dsp
setmqaut -m QMgrName -n ** -t authinfo -g GroupName +dsp
setmqaut -m QMgrName -n ** -t listener -g GroupName +dsp
setmqaut -m QMgrName -n ** -t namelist -g GroupName +dsp
setmqaut -m QMgrName -n ** -t process -g GroupName +dsp
setmqaut -m QMgrName -n ** -t service -g GroupName +dsp
setmqaut -m QMgrName -t qmgr -g GroupName +dsp +inq +connect
```

  Droits d'accès spécifiques à SYSTEM.ADMIN.COMMAND.QUEUE et SYSTEM.MQEXPLORER.REPLY.MODEL n'est nécessaire que si vous souhaitez utiliser le IBM MQ Explorer.
- IBM i
  Pour IBM i, exécutez les commandes suivantes:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*Q) USER('GroupName') AUT(*ADMDSP *BROWSE) MQMNAME('QMgrName')
```

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*TOPIC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CHL) USER('GroupName') AUT(*ADMDSP *INQ) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CLTCN) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*AUTHINFO) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*LSR) USER('GroupName') AUT(*ADMDSP)MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*NMLIST) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*SVC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ('object-name') OBJTYPE(*MQM) USER('GroupName') AUT(*ADMDSP *CONNECT *INQ)
MQMNAME('QMgrName')
```

- z/OS
  Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQQUEUE QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQQUEUE) ID(GroupName) ACCESS(READ)
RDEFINE MXTOPIC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MXTOPIC) ID(GroupName) ACCESS(READ)
RDEFINE MQPROC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQPROC) ID(GroupName) ACCESS(READ)
RDEFINE MQNLIST QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi d'un accès administrateur complet à toutes les ressources d'un gestionnaire de files d'attente

Accordez un accès administrateur complet à toutes les ressources d'un gestionnaire de files d'attente, à chaque utilisateur ou groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser l'assistant Ajouter des droits basés sur les rôles ou les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Remarques : ULW

1. Si vous utilisez **runmqsc** pour administrer le gestionnaire de files d'attente à la place de IBM MQ Explorer, vous devez accorder les droits permettant d'interroger, d'obtenir et de parcourir SYSTEM.MQSC.REPLY.QUEUE, et vous n'avez pas besoin d'accorder des droits sur SYSTEM.MQEXPLORER.REPLY.MODEL .
2. Lorsque vous octroyez à un utilisateur l'accès à toutes les ressources d'un gestionnaire de files d'attente, il existe des commandes que l'utilisateur ne peut pas exécuter, à moins qu'il ne dispose d'un accès en lecture au fichier `qm.ini` . Cela est dû aux restrictions qui s'appliquent aux utilisateurs non `mqm` qui peuvent lire le fichier `qm.ini` .

   L'utilisateur ne peut pas exécuter les commandes suivantes sauf si vous lui avez accordé un accès en lecture au fichier qm.ini :

   - Définition d'un canal configuré pour utiliser TLS
   - Définition d'un canal à l'aide de variables d'insertion de configuration automatique définies dans `qm.ini`

Procédure

- Si vous utilisez l'assistant, dans le panneau IBM MQ Explorer Navigator , cliquez avec le bouton droit de la souris sur le gestionnaire de files d'attente, puis cliquez sur **Droits sur les objets** > **Ajouter des droits basés sur les rôles**.

  L'assistant Ajout de droits basés sur des rôles s'ouvre.

- Linux UNIX

  Pour les systèmes UNIX and Linux , exécutez les commandes suivantes:

```
setmqaut -m QMgrName -n '**' -t queue -g GroupName +alladm +browse
setmqaut -m QMgrName -n @class -t queue -g GroupName +crt
setmqaut -m QMgrName -n SYSTEM.ADMIN.COMMAND.QUEUE -t queue -g GroupName +dsp +inq +put
setmqaut -m QMgrName -n SYSTEM.MQEXPLORER.REPLY.MODEL -t queue -g GroupName +dsp +inq +get +put
setmqaut -m QMgrName -n '**' -t topic -g GroupName +alladm
setmqaut -m QMgrName -n @class -t topic -g GroupName +crt
setmqaut -m QMgrName -n '**' -t channel -g GroupName +alladm
setmqaut -m QMgrName -n @class -t channel -g GroupName +crt
setmqaut -m QMgrName -n '**' -t clntconn -g GroupName +alladm
setmqaut -m QMgrName -n @class -t clntconn -g GroupName +crt
setmqaut -m QMgrName -n '**' -t authinfo -g GroupName +alladm
setmqaut -m QMgrName -n @class -t authinfo -g GroupName +crt
setmqaut -m QMgrName -n '**' -t listener -g GroupName +alladm
setmqaut -m QMgrName -n @class -t listener -g GroupName +crt
setmqaut -m QMgrName -n '**' -t namelist -g GroupName +alladm
setmqaut -m QMgrName -n @class -t namelist -g GroupName +crt
setmqaut -m QMgrName -n '**' -t process -g GroupName +alladm
setmqaut -m QMgrName -n @class -t process -g GroupName +crt
setmqaut -m QMgrName -n '**' -t service -g GroupName +alladm
setmqaut -m QMgrName -n @class -t service -g GroupName +crt
setmqaut -m QMgrName -t qmgr -g GroupName +alladm +connect
```

  Voir **setmqaut** pour plus d'informations sur `@class`

- Windows

  Pour les systèmes Windows , exécutez les mêmes commandes que pour les systèmes UNIX and Linux , mais en utilisant le nom de profil @CLASS au lieu de @class.

- IBM i

  Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*ALL) USER(' GroupName ') AUT(*ALLADM) MQMNAME(' QMgrName ')
```

- z/OS

  Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQADMIN QMgrName.*.** UACC(NONE)
PERMIT QMgrName.*.** CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

  Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente.

z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Suppression de la connectivité au gestionnaire de files d'attente

Si vous ne souhaitez pas que les applications utilisateur se connectent à votre gestionnaire de files d'attente, supprimez leurs droits de connexion.

Pourquoi et quand exécuter cette tâche

Révoquez le droit de tous les utilisateurs de se connecter au gestionnaire de files d'attente à l'aide de la commande appropriée pour votre système d'exploitation.

Sur les systèmes UNIX, Linux, Windows et IBM i, vous pouvez également utiliser la commande DELETE AUTHREC .

Remarque : Sur IBM MQ Appliance , vous ne pouvez utiliser que la commande **DELETE AUTHREC** .

Procédure

- ULW
  Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -t qmgr -g GroupName -connect
```

- IBM i
  Pour IBM i, exécutez la commande suivante:

```
RVKMQMAUT OBJ ('QMgrName') OBJTYPE(*MQM) USER(*ALL) AUT(*CONNECT)
```

- z/OS
  Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
```

  N'émettez aucune commande PERMIT.

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente.

  z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  GroupName
  Nom du groupe auquel l'accès doit être refusé.

Autorisation des applications utilisateur à se connecter à votre gestionnaire de files d'attente

Vous souhaitez autoriser l'application utilisateur à se connecter à votre gestionnaire de files d'attente. Utilisez les tableaux de cette rubrique pour déterminer les actions à effectuer.

Tout d'abord, déterminez si les applications client se connecteront à votre gestionnaire de files d'attente.

Si aucune des applications qui se connecteront à votre gestionnaire de files d'attente ne sont des applications client, désactivez l'accès distant comme décrit dans «Désactivation de l'accès distant au gestionnaire de files d'attente», à la page 404.

Si une ou plusieurs des applications qui se connecteront à votre gestionnaire de files d'attente sont des applications client, sécurisez la connectivité à distance comme décrit dans «Sécurisation de la connectivité distante au gestionnaire de files d'attente», à la page 396.

Dans les deux cas, configurez la sécurité de connexion comme décrit dans «Configuration de la sécurité de connexion», à la page 404

Si vous souhaitez contrôler l'accès aux ressources pour chaque utilisateur se connectant au gestionnaire de files d'attente, voir le tableau suivant. Si l'instruction de la première colonne est vraie, effectuez l'action indiquée dans la deuxième colonne.

Instruction	Effectuez cette action
Vous disposez d'applications qui utilisent des files d'attente	Pour plus d'informations, voir «Contrôle de l'accès utilisateur aux files d'attente», à la page 405.
Vous disposez d'applications qui utilisent des rubriques	Voir «Contrôle de l'accès des utilisateurs aux rubriques», à la page 412.
Vous disposez d'applications qui s'interrogent sur l'objet gestionnaire de files d'attente	Voir «Octroi de droits d'interrogation sur un gestionnaire de files d'attente», à la page 414.
Vous disposez d'applications qui utilisent des objets de processus	Pour plus d'informations, voir «Octroi de droits d'accès aux processus», à la page 415.
Vous disposez d'applications qui utilisent des listes de noms	Pour plus d'informations, voir «Octroi de droits d'accès aux listes de noms», à la page 415.

Sécurisation de la connectivité distante au gestionnaire de files d'attente

Vous pouvez sécuriser la connectivité distante au gestionnaire de files d'attente à l'aide de TLS, d'un exit de sécurité, d'enregistrements d'authentification de canal ou d'une combinaison de ces méthodes.

Pourquoi et quand exécuter cette tâche

Vous connectez un client au gestionnaire de files d'attente à l'aide d'un canal de connexion client sur le poste de travail client et d'un canal de connexion serveur sur le serveur. Sécurisez ces connexions de l'une des manières suivantes.

Procédure

1. Utilisation de TLS avec des enregistrements d'authentification de canal:
   a) Empêchez tout nom distinctif (DN) d'ouvrir un canal en utilisant un enregistrement d'authentification de canal SSLPEERMAP pour mapper tous les noms distinctifs à USERSRC (NOACCESS).
   b) Autoriser des noms distinctifs ou des ensembles de noms distinctifs spécifiques à ouvrir un canal à l'aide d'un enregistrement d'authentification de canal SSLPEERMAP pour les mapper à USERSRC (CHANNEL).
2. Utilisation de TLS avec un exit de sécurité:

a) Définissez MCAUSER sur le canal de connexion serveur sur un identificateur utilisateur sans privilèges.

b) Ecrivez un exit de sécurité pour affecter une valeur MCAUSER en fonction de la valeur du nom distinctif TLS qu'il reçoit dans les zones SSLPeerNamePtr et SSLPeerNameLength transmises à l'exit dans la structure MQCD.

3. Utilisation de TLS avec des valeurs de définition de canal fixes:
   a) Définissez SSLPEER sur le canal de connexion serveur sur une valeur spécifique ou une plage de valeurs étroite.
   b) Définissez MCAUSER sur le canal de connexion serveur sur l'ID utilisateur avec lequel le canal doit s'exécuter.
4. Utilisation des enregistrements d'authentification de canal sur les canaux qui n'utilisent pas TLS:
   a) Empêchez toute adresse IP d'ouvrir des canaux en utilisant un enregistrement d'authentification de canal de mappage d'adresse avec ADDRESS (*) et USERSRC (NOACCESS).
   b) Autorisez des adresses IP spécifiques à ouvrir des canaux, en utilisant des enregistrements d'authentification de canal de mappage d'adresse pour ces adresses avec USERSRC (CHANNEL).
5. A l'aide d'un exit de sécurité:
   a) Ecrivez un exit de sécurité pour autoriser les connexions en fonction de la propriété que vous choisissez, par exemple, l'adresse IP d'origine.
6. Il est également possible d'utiliser des enregistrements d'authentification de canal avec un exit de sécurité ou d'utiliser les trois méthodes, si vos circonstances particulières l'exigent.

Blocage d'adresses IP spécifiques
Vous pouvez empêcher un canal spécifique d'accepter une connexion entrante à partir d'une adresse IP ou empêcher l'ensemble du gestionnaire de files d'attente d'autoriser l'accès à partir d'une adresse IP, à l'aide d'un enregistrement d'authentification de canal.

Avant de commencer

Activez les enregistrements d'authentification de canal en exécutant la commande suivante:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Pourquoi et quand exécuter cette tâche

Pour empêcher des canaux spécifiques d'accepter une connexion entrante et s'assurer que les connexions sont uniquement acceptées lors de l'utilisation du nom de canal correct, un type de règle peut être utilisé pour bloquer les adresses IP. Pour interdire l'accès d'une adresse IP à l'ensemble du gestionnaire de files d'attente, vous devez normalement utiliser un pare-feu pour le bloquer définitivement. Toutefois, un autre type de règle peut être utilisé pour vous permettre de bloquer temporairement quelques adresses, par exemple pendant que vous attendez que le pare-feu soit mis à jour.

Procédure

- Pour empêcher les adresses IP d'utiliser un canal spécifique, définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**.

```
SET CHLAUTH(generic-channel-name) TYPE(ADDRESSMAP) ADDRESS(generic-ip-address)
USERSRC(NOACCESS)
```

La commande comporte trois parties:

SET CHLAUTH (*nom-canal-générique*)
Cette partie de la commande permet de contrôler si vous souhaitez bloquer une connexion pour l'ensemble du gestionnaire de files d'attente, un canal unique ou une plage de canaux. Ce que vous mettez ici détermine les zones qui sont couvertes.

Exemple :

- SET CHLAUTH('*') -bloque tous les canaux d'un gestionnaire de files d'attente, c'est-à-dire l'intégralité du gestionnaire de files d'attente
- SET CHLAUTH ('SYSTEM. *')-bloque chaque canal commençant par SYSTEM.
- SET CHLAUTH ('SYSTEM.DEF.SVRCONN')-bloque le canal SYSTEM.DEF.SVRCONN

Type de règle CHLAUTH
Utilisez cette partie de la commande pour spécifier le type de commande et déterminer si vous souhaitez fournir une adresse unique ou une liste d'adresses.

Exemple :

- TYPE(ADDRESSMAP) -Utilisez ADDRESSMAP si vous souhaitez fournir une adresse unique ou une adresse générique. Par exemple, ADDRESS('192.168.*') bloque toutes les connexions provenant d'une adresse IP à partir de 192.168.

  Pour plus d'informations sur le filtrage des adresses IP à l'aide de modèles, voir Adresses IP génériques.
- TYPE(BLOCKADDR) -Utilisez BLOCKADDR si vous souhaitez fournir une liste d'adresses à bloquer.

Paramètres supplémentaires
Ces paramètres dépendent du type de règle que vous avez utilisé dans la deuxième partie de la commande:

- Pour TYPE(ADDRESSMAP) , vous utilisez ADDRESS
- Pour TYPE(BLOCKADDR) , vous utilisez ADDRLIST

Référence associée
SET CHLAUTH

Blocage temporaire d'adresses IP spécifiques si le gestionnaire de files d'attente n'est pas en cours d'exécution
Vous pouvez bloquer des adresses IP particulières ou des plages d'adresses lorsque le gestionnaire de files d'attente n'est pas en cours d'exécution et que vous ne pouvez donc pas émettre de commandes MQSC. Vous pouvez temporairement bloquer des adresses IP de manière exceptionnelle en modifiant le fichier blockaddr.ini .

Pourquoi et quand exécuter cette tâche

Le fichier blockaddr.ini contient une copie des définitions BLOCKADDR utilisées par le gestionnaire de files d'attente. Ce fichier est lu par le programme d'écoute si ce dernier est démarré avant le gestionnaire de files d'attente. Dans ces circonstances, le programme d'écoute utilise toutes les valeurs que vous avez ajoutées manuellement au fichier blockaddr.ini .

Toutefois, sachez que lorsque le gestionnaire de files d'attente est démarré, il écrit l'ensemble des définitions BLOCKADDR dans le fichier blockaddr.ini , en écrase les modifications manuelles que vous avez éventuellement effectuées. De même, chaque fois que vous ajoutez ou supprimez une définition BLOCKADDR à l'aide de la commande **SET CHLAUTH** , le fichier blockaddr.ini est mis à jour. Vous pouvez donc apporter des modifications permanentes aux définitions BLOCKADDR uniquement à l'aide de la commande **SET CHLAUTH** lorsque le gestionnaire de files d'attente est en cours d'exécution.

Procédure

1. Ouvrez le fichier blockaddr.ini dans n'importe quel éditeur de texte.

Le fichier se trouve dans le répertoire de données du gestionnaire de files d'attente.

2. Ajoutez des adresses IP sous forme de paires mot clé-valeur simples, où le mot clé est `Addr`.

   Pour plus d'informations sur le filtrage des adresses IP à l'aide de modèles, voir Adresses IP génériques.

   Exemple :

```
Addr = 192.0.2.0
Addr = 192.0.*
Addr = 192.0.2.1-8
```

Tâches associées

«Blocage d'adresses IP spécifiques», à la page 397
Vous pouvez empêcher un canal spécifique d'accepter une connexion entrante à partir d'une adresse IP ou empêcher l'ensemble du gestionnaire de files d'attente d'autoriser l'accès à partir d'une adresse IP, à l'aide d'un enregistrement d'authentification de canal.

Référence associée

SET CHLAUTH

Blocage d'ID utilisateur spécifiques

Vous pouvez empêcher des utilisateurs spécifiques d'utiliser un canal en spécifiant des ID utilisateur qui, s'ils sont vérifiés, provoquent l'arrêt du canal. Pour cela, définissez un enregistrement d'authentification de canal.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procédure

Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(BLOCKUSER) USERLIST(userID1, userID2)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.

La liste d'utilisateurs fournie sur un TYPE(BLOCKUSER) s'applique uniquement aux canaux SVRCONN et non aux canaux de gestionnaire de files d'attente.

userID1 et *userID2* sont chacun l'ID d'un utilisateur qui doit être empêché d'utiliser le canal. Vous pouvez également spécifier la valeur spéciale *MQADMIN pour faire référence aux administrateurs privilégiés. Pour plus d'informations sur les utilisateurs privilégiés, voir «Utilisateurs privilégiés», à la page 343. Pour plus d'informations sur *MQADMIN, voir SET CHLAUTH.

Référence associée

SET CHLAUTH

Mappage d'un gestionnaire de files d'attente éloignées à un ID utilisateur MCAUSER

Vous pouvez utiliser un enregistrement d'authentification de canal pour définir l'attribut MCAUSER d'un canal, en fonction du gestionnaire de files d'attente à partir duquel le canal se connecte.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Pourquoi et quand exécuter cette tâche

Vous pouvez éventuellement restreindre les adresses IP auxquelles la règle s'applique.

Notez que cette technique ne s'applique pas aux canaux de connexion serveur. Si vous spécifiez le nom d'un canal de connexion serveur dans les commandes suivantes, cela n'a aucun effet.

Procédure

- Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

  ```
  SET CHLAUTH(' generic-channel-name ') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
  ) USERSRC(MAP) MCAUSER(user)
  ```

  nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
  generic-partner-qmgr-name est soit le nom du gestionnaire de files d'attente, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du gestionnaire de files d'attente.
  user est l'ID utilisateur à utiliser pour toutes les connexions à partir du gestionnaire de files d'attente spécifié.
- Pour limiter cette commande à certaines adresses IP, incluez le paramètre **ADDRESS** comme suit:

  ```
  SET CHLAUTH(' generic-channel-name ') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
  ) USERSRC(MAP) MCAUSER(user) ADDRESS(
  generic-ip-address)
  ```

  nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
  generic-ip-address est soit une adresse unique, soit un modèle incluant l'astérisque (*) comme caractère générique ou le trait d'union (-) pour indiquer une plage, qui correspond à l'adresse. Pour plus d'informations sur les adresses IP génériques, voir Adresses IP génériques.

Référence associée
SET CHLAUTH

Mappage d'un ID utilisateur client à un ID utilisateur MCAUSER
Vous pouvez utiliser un enregistrement d'authentification de canal pour modifier l'attribut MCAUSER d'un canal de connexion serveur, en fonction de l'ID utilisateur reçu d'un client.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Pourquoi et quand exécuter cette tâche

Notez que cette technique s'applique uniquement aux canaux de connexion serveur. Il n'a aucun effet sur les autres types de canal.

Procédure

Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record** . Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE (USERMAP) CLNTUSER(client-user-name) USERSRC(MAP)
MCAUSER(
user)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
client-user-name est l'ID utilisateur associé à la connexion client. La valeur peut être vérifiée par l'application client, modifiée par l'authentification de connexion à l'aide de l'adoption anticipée ou définie via un exit de canal.
user est l'ID utilisateur à utiliser à la place du nom d'utilisateur du client.

Référence associée
SET CHLAUTH
Attributs de la strophe channels (ChlauthEarlyAdopt)

Mappage d'un nom distinctif SSL ou TLS à un ID utilisateur MCAUSER
Vous pouvez utiliser un enregistrement d'authentification de canal pour définir l'attribut MCAUSER d'un canal, en fonction du nom distinctif (DN) reçu.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procédure

Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (SSLPEERMAP)
SSLPEER(generic-ssl-peer-name) SSLCERTI(generic-issuer-name)
USERSRC(MAP) MCAUSER(user)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
generic-ssl-peer-name est une chaîne qui suit les règles IBM MQ standard pour les valeurs SSLPEER. Voir Règles IBM MQ pour les valeurs SSLPEER.
user est l'ID utilisateur à utiliser pour toutes les connexions utilisant le nom distinctif spécifié.
generic-issuer-name fait référence au nom distinctif de l'émetteur du certificat à mettre en correspondance. Ce paramètre est facultatif, mais vous devez l'utiliser afin d'éviter toute correspondance avec le mauvais certificat, si plusieurs autorités de certification sont utilisées.

Référence associée
SET CHLAUTH

Blocage de l'accès à partir d'un gestionnaire de files d'attente éloignées
Vous pouvez utiliser un enregistrement d'authentification de canal pour empêcher un gestionnaire de files d'attente éloignées de démarrer des canaux.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Pourquoi et quand exécuter cette tâche

Notez que cette technique ne s'applique pas aux canaux de connexion serveur. Si vous indiquez le nom d'un canal de connexion serveur dans la commande suivante, cela n'a aucun effet.

Procédure

Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(QMGRMAP) QMNAME(' generic-partner-qmgr-name ')
USERSRC(NOACCESS)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
generic-partner-qmgr-name est soit le nom du gestionnaire de files d'attente, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du gestionnaire de files d'attente.

Référence associée

SET CHLAUTH

Blocage de l'accès pour un ID utilisateur client

Vous pouvez utiliser un enregistrement d'authentification de canal pour empêcher un ID utilisateur client d'établir une connexion de canal.

Avant de commencer

Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Pourquoi et quand exécuter cette tâche

Notez que cette technique s'applique uniquement aux canaux de connexion serveur. Il n'a aucun effet sur les autres types de canal.

Procédure

Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(USERMAP) CLNTUSER(' client-user-name ')
USERSRC(NOACCESS)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
client-user-name est l'ID utilisateur associé à la connexion client. La valeur peut être vérifiée par l'application client, modifiée par l'authentification de connexion à l'aide de l'adoption anticipée ou définie via un exit de canal.

Référence associée
SET CHLAUTH

Blocage de l'accès pour un nom distinctif SSL ou TLS
Vous pouvez utiliser un enregistrement d'authentification de canal pour empêcher un nom distinctif TLS de démarrer des canaux.

Avant de commencer
Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procédure
Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(SSLPEERMAP)
SSLPEER('generic-ssl-peer-name') SSLCERTI(generic-issuer-name)
USERSRC(NOACCESS)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
generic-ssl-peer-name est une chaîne qui suit les règles IBM MQ standard pour les valeurs SSLPEER. Voir Règles IBM MQ pour les valeurs SSLPEER.
generic-issuer-name fait référence au nom distinctif de l'émetteur du certificat à mettre en correspondance. Ce paramètre est facultatif, mais vous devez l'utiliser afin d'éviter toute correspondance avec le mauvais certificat, si plusieurs autorités de certification sont utilisées.

Référence associée
SET CHLAUTH

Mappage d'une adresse IP à un ID utilisateur MCAUSER
Vous pouvez utiliser un enregistrement d'authentification de canal pour définir l'attribut MCAUSER d'un canal, en fonction de l'adresse IP à partir de laquelle la connexion est reçue.

Avant de commencer
Vérifiez que les enregistrements d'authentification de canal sont activés comme suit:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procédure
Définissez un enregistrement d'authentification de canal à l'aide de la commande MQSC **SET CHLAUTH**ou de la commande PCF **Set Channel Authentication Record**. Par exemple, vous pouvez émettre la commande MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(ADDRESSMAP) ADDRESS(' generic-ip-address ')
USERSRC(MAP) MCAUSER(user)
```

nom-canal-générique est soit le nom d'un canal auquel vous souhaitez contrôler l'accès, soit un modèle incluant l'astérisque (*) comme caractère générique qui correspond au nom du canal.
user est l'ID utilisateur à utiliser pour toutes les connexions utilisant le nom distinctif spécifié.

generic-ip-address est soit l'adresse à partir de laquelle la connexion est établie, soit un modèle incluant l'astérisque (*) comme caractère générique ou le trait d'union (-) pour indiquer une plage qui correspond à l'adresse.

Référence associée
SET CHLAUTH

Désactivation de l'accès distant au gestionnaire de files d'attente
Si vous ne souhaitez pas que les applications client se connectent à votre gestionnaire de files d'attente, désactivez l'accès à distance à ce dernier.

Pourquoi et quand exécuter cette tâche
Empêchez les applications client de se connecter au gestionnaire de files d'attente de l'une des manières suivantes:

Procédure

- Supprimez tous les canaux de connexion serveur à l'aide de la commande MQSC **DELETE CHANNEL**.
- Définissez l'ID utilisateur de l'agent de canal de transmission de messages (MCAUSER) du canal sur un ID utilisateur sans droits d'accès, à l'aide de la commande MQSC **ALTER CHANNEL**.

Configuration de la sécurité de connexion
Accordez le droit de se connecter au gestionnaire de files d'attente à chaque utilisateur ou groupe d'utilisateurs dont l'entreprise a besoin pour le faire.

Pourquoi et quand exécuter cette tâche
Pour configurer la sécurité de la connexion, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Sous UNIX, Linux, and Windows :

  ```
  setmqaut -m QMgrName -t qmgr -g GroupName +connect
  ```

- IBM i
  Sous IBM i :

  ```
  GRTMQMAUT OBJ('QMgrName') OBJTYPE(*MQM) USER('GroupName') AUT(*CONNECT)
  ```

- z/OS
  Sous z/OS :

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

Ces commandes donnent le droit de se connecter pour le traitement par lots, CICS, IMS et l'initiateur de canal (CHIN). Si vous n'utilisez pas un type particulier de connexion, omettez les commandes appropriées.

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Concepts associés

«Profils de sécurité de connexion pour l'initiateur de canal», à la page 202
Les profils de vérification des connexions à partir de l'initiateur de canal sont composés du nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente suivi du mot *CHIN*. Accordez à l'ID utilisateur utilisé par l'espace adresse de la tâche démarrée de l'initiateur de canal l'accès en lecture (READ) au profil de connexion.

Contrôle de l'accès utilisateur aux files d'attente

Vous souhaitez contrôler l'accès des applications aux files d'attente. Cette rubrique permet de déterminer les actions à effectuer.

Pour chaque instruction true de la première colonne, effectuez l'action indiquée dans la deuxième colonne.

Instruction	Action
L'application extrait des messages d'une file d'attente	Pour plus d'informations, voir «Octroi de droits pour l'obtention de messages à partir de files d'attente», à la page 406.
L'application définit le contexte	Pour plus d'informations, voir «Octroi de droits d'accès pour définir le contexte», à la page 406.
L'application transmet le contexte	Pour plus d'informations, voir «Octroi de l'autorisation de transmettre le contexte», à la page 408.
L'application insère des messages dans une file d'attente en cluster	Pour plus d'informations, voir «Autorisation d'insertion de messages dans des files d'attente de cluster éloignées», à la page 474.
L'application insère des messages dans une file d'attente locale	Pour plus d'informations, voir «Octroi du droit d'insertion de messages dans une file d'attente locale», à la page 409.
L'application insère des messages dans une file d'attente modèle	Pour plus d'informations, voir «Octroi du droit d'insertion de messages dans une file d'attente modèle», à la page 410.

Instruction	Action
L'application insère des messages dans une file d'attente éloignée	Pour plus d'informations, voir «Octroi du droit d'insertion de messages dans une file d'attente de cluster éloignée», à la page 410.

Octroi de droits pour l'obtention de messages à partir de files d'attente
Accordez le droit d'extraire des messages d'une file d'attente ou d'un ensemble de files d'attente à chaque groupe d'utilisateurs qui en a besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit d'extraire des messages de certaines files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +get
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*GET) MQMNAME(' QMgrName ')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(UPDATE)
```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

Octroi de droits d'accès pour définir le contexte
Accordez le droit de définir le contexte d'un message en cours d'insertion à chaque groupe d'utilisateurs qui en ont besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit de définir le contexte sur certaines files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i
- Linux
- UNIX
- Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez l'une des commandes suivantes:
  - Pour définir le contexte d'identité uniquement:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setid
```

  - Pour définir tous les contextes:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setall
```

  Remarque : Pour utiliser les droits `setid` ou `setall` , les autorisations doivent être accordées à la fois sur l'objet de file d'attente approprié et sur l'objet de gestionnaire de files d'attente.
- Pour IBM i, exécutez l'une des commandes suivantes:
  - Pour définir le contexte d'identité uniquement:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*SETID) MQMNAME('
QMgrName ')
```

  - Pour définir tous les contextes:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*SETALL) MQMNAME('
QMgrName ')
```

- Pour z/OS, exécutez l'un des ensembles de commandes suivants:
  - Pour définir le contexte d'identité uniquement:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  - Pour définir tous les contextes:

```
RDEFINE MQQUEUE QMgrName. ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(CONTROL)
```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi de l'autorisation de transmettre le contexte
Accordez le droit de transmettre le contexte d'un message extrait à un message en cours d'insertion, à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit de transmettre le contexte sur certaines files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- ULW
  Pour les systèmes UNIX, Linux, and Windows , exécutez l'une des commandes suivantes:
  - Pour transmettre uniquement le contexte d'identité:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passid
    ```

  - Pour transmettre tous les contextes:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passall
    ```

- IBM i
  Pour IBM i, exécutez l'une des commandes suivantes:
  - Pour transmettre uniquement le contexte d'identité:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PASSID) MQMNAME('
    QMgrName ')
    ```

  - Pour transmettre tous les contextes:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PASSALL) MQMNAME('
    QMgrName ')
    ```

- z/OS
  Pour z/OS, exécutez les commandes suivantes pour transmettre le contexte d'identité ou tout le contexte:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi du droit d'insertion de messages dans une file d'attente locale
Accordez le droit d'insérer des messages dans une file d'attente locale ou dans un ensemble de files d'attente, à chaque groupe d'utilisateurs qui en a besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit d'insertion de messages dans certaines files d'attente locales, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i
- Linux
- UNIX
- IBM i

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi du droit d'insertion de messages dans une file d'attente modèle
Accordez le droit d'insérer des messages dans une file d'attente modèle ou dans un ensemble de files d'attente modèle, à chaque groupe d'utilisateurs qui en a besoin.

Pourquoi et quand exécuter cette tâche

Les files d'attente modèles sont utilisées pour créer des files d'attente dynamiques. Vous devez donc accorder des droits d'accès au modèle et aux files d'attente dynamiques. Pour accorder ces droits, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez les commandes suivantes:

```
setmqaut -m QMgrName -n ModelQueueName -t queue -g GroupName +put
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Pour IBM i, exécutez les commandes suivantes:

```
GRTMQMAUT OBJ(' ModelQueueName ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQQUEUE QMgrName.ModelQueueName UACC(NONE)
PERMIT QMgrName.ModelQueueName CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  Nom ModelQueue
  Nom de la file d'attente modèle sur laquelle sont basées les files d'attente dynamiques.

  ObjectProfile
  Nom de la file d'attente dynamique ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

Octroi du droit d'insertion de messages dans une file d'attente de cluster éloignée
Accordez le droit d'insérer des messages dans une file d'attente de cluster éloignée ou dans un ensemble de files d'attente, à chaque groupe d'utilisateurs qui en a besoin.

Pourquoi et quand exécuter cette tâche

Pour placer un message dans une file d'attente de cluster éloignée, vous pouvez le placer dans une définition locale d'une file d'attente éloignée ou dans une file d'attente éloignée qualifiée complète. Si vous utilisez une définition locale d'une file d'attente éloignée, vous devez disposer des droits d'accès à l'objet local: voir «Octroi du droit d'insertion de messages dans une file d'attente locale», à la page 409. Si vous utilisez une file d'attente éloignée qualifiée complète, vous devez disposer des droits nécessaires pour la placer dans la file d'attente éloignée. Accordez ces droits à l'aide des commandes appropriées pour votre système d'exploitation.

Le comportement par défaut consiste à effectuer un contrôle d'accès sur le `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Notez que ce comportement s'applique, même si vous utilisez plusieurs files d'attente de transmission.

Le comportement spécifique décrit dans cette rubrique s'applique uniquement lorsque vous avez configuré l'attribut **`ClusterQueueAccessControl`** dans le fichier `qm.ini` comme étant *RQMName*, comme décrit dans la rubrique Strophe de sécurité , puis redémarré le gestionnaire de files d'attente.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

  ```
  setmqaut -m QMgrName -t rqmname -n
  ObjectProfile -g GroupName +put
  ```

  Notez que vous pouvez utiliser l'objet *rqmname* uniquement pour les files d'attente de cluster éloignées.
- Pour IBM i, exécutez la commande suivante:

  ```
  GRTMQMAUT OBJTYPE(*RMTMQMNAME) OBJ('
  ObjectProfile') USER(GroupName) AUT(*PUT) MQMNAME('
  QMgrName')
  ```

  Notez que vous pouvez utiliser l'objet RMTMQMNAME uniquement pour les files d'attente de cluster éloignées.
- Pour z/OS, exécutez les commandes suivantes:

  ```
  RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
  PERMIT QMgrName.QUEUE.ObjectProfile CLASS(MQQUEUE)
  ID(GroupName) ACCESS(UPDATE)
  ```

  Notez que vous pouvez utiliser le nom du gestionnaire de files d'attente éloignées (ou du groupe de partage de files d'attente) uniquement pour les files d'attente de cluster éloignées.

  Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom du gestionnaire de files d'attente éloignées ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Contrôle de l'accès des utilisateurs aux rubriques

Vous devez contrôler l'accès des applications aux rubriques. Cette rubrique permet de déterminer les actions à effectuer.

Pour chaque instruction true de la première colonne, effectuez l'action indiquée dans la deuxième colonne.

Tableau 71. Contrôle de l'accès des utilisateurs aux rubriques

Instruction	**Action**
L'application publie des messages dans un sujet	Pour plus d'informations, voir «Octroi du droit de publier des messages dans une rubrique», à la page 412.
L'application s'abonne à une rubrique	Pour plus d'informations, voir «Octroi de droits d'abonnement à des rubriques», à la page 413.

Octroi du droit de publier des messages dans une rubrique

Accordez le droit de publier des messages dans une rubrique ou un ensemble de rubriques, à chaque groupe d'utilisateurs qui en a besoin.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit de publier des messages dans certaines rubriques, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +pub
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(*PUB) MQMNAME('
QMgrName ')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQTOPIC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi de droits d'abonnement à des rubriques
Accordez le droit de s'abonner à une rubrique ou à un ensemble de rubriques, à chaque groupe d'utilisateurs ayant besoin d'une rubrique ou d'un ensemble de rubriques.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit d'abonnement à certaines rubriques, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i
- Linux
- UNIX
- IBM i

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +sub
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(*SUB) MQMNAME('
QMgrName ')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQTOPIC QMgrName.SUBSCRIBE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SUBSCRIBE.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi de droits d'interrogation sur un gestionnaire de files d'attente

Accordez les droits d'interrogation sur un gestionnaire de files d'attente à chaque groupe d'utilisateurs ayant besoin d'un gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit d'interrogation sur un gestionnaire de files d'attente, utilisez les commandes appropriées pour votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i
- Linux
- UNIX
- Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName +inq
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(*INQ) MQMNAME(' QMgrName
')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQCMDS QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Ces commandes permettent d'accéder au gestionnaire de files d'attente spécifié. Pour permettre à l'utilisateur d'utiliser la commande MQINQ, exécutez les commandes suivantes:

```
RDEFINE MQCMDS QMgrName.MQINQ.QMGR UACC(NONE)
PERMIT QMgrName.MQINQ.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

Octroi de droits d'accès aux processus

Accordez le droit d'accès à un processus ou à un ensemble de processus à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder le droit d'accès à certains processus, utilisez les commandes appropriées à votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +all
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(*ALL) MQMNAME(' QMgrName
')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQPROC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQPROC) ID(GroupName) ACCESS(READ)
```

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  ObjectProfile
  Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

Octroi de droits d'accès aux listes de noms

Accordez le droit d'accéder à une liste de noms ou à un ensemble de listes de noms, à chaque groupe d'utilisateurs ayant un besoin métier.

Pourquoi et quand exécuter cette tâche

Pour accorder les droits d'accès à certaines listes de noms, utilisez les commandes appropriées à votre système d'exploitation.

Sur les plateformes suivantes, vous pouvez également utiliser la commande SET AUTHREC :

- IBM i IBM i

- Linux Linux
- UNIX UNIX
- IBM i Windows

Remarque : MQ Appliance Sous IBM MQ Appliance , vous ne pouvez utiliser que la commande **SET AUTHREC** .

Procédure

- Pour les systèmes UNIX, Linux, and Windows , exécutez la commande suivante:

```
setmqaut -m QMgrName -n
ObjectProfile -t namelist -g GroupName
+all
```

- Pour IBM i, exécutez la commande suivante:

```
GRTMQMAUT OBJ('ObjectProfile
') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALL) MQMNAME('
QMgrName')
```

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQNLIST
QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile
CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
```

Les noms de variable ont les significations suivantes:

QMgrName
Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

ObjectProfile
Nom de l'objet ou du profil générique pour lequel modifier les autorisations.

GroupName
Nom du groupe auquel l'accès doit être accordé.

ULW Droit d'administration de IBM MQ sur UNIX, Linux, and Windows

Les administrateurs IBM MQ peuvent utiliser toutes les commandes IBM MQ et accorder des droits à d'autres utilisateurs. Lorsque les administrateurs émettent des commandes pour les gestionnaires de files d'attente éloignées, ils doivent disposer des droits requis sur le gestionnaire de files d'attente éloignées. D'autres considérations s'appliquent aux systèmes Windows .

Les administrateurs IBM MQ ont le droit d'utiliser toutes les commandes IBM MQ (y compris les commandes permettant d'accorder des droits IBM MQ à d'autres utilisateurs).

Pour être un administrateur IBM MQ , vous devez être membre d'un groupe spécial appelé groupe **mqm** .

Windows Sinon, sous Windows uniquement, les comptes locaux peuvent administrer IBM MQ s'ils sont membres du groupe Administrateurs sur les systèmes Windows .

Avertissement : Vous pouvez ajouter votre utilisateur Azure AD au groupe mqm à l'aide d'une commande d'administrateur. Par exemple, utilisez la commande `net localgroup mqm AzureAD\<your userID> /add`. Exécutez ensuite les commandes d'administration IBM MQ ou utilisez IBM MQ Explorer.

Le groupe **mqm** est créé automatiquement lorsque IBM MQ est installé. Vous pouvez ajouter d'autres utilisateurs au groupe pour leur permettre d'effectuer des tâches d'administration. Tous les membres de ce groupe peuvent accéder à toutes les ressources. Cet accès peut être révoqué uniquement en supprimant un utilisateur du groupe **mqm** et en exécutant la commande **REFRESH SECURITY** .

Les administrateurs peuvent utiliser des commandes de contrôle pour administrer IBM MQ. L'une de ces commandes de contrôle est **setmqaut**, qui est utilisée pour accorder des droits à d'autres utilisateurs afin de leur permettre d'accéder aux ressources IBM MQ ou de les contrôler. Les commandes PCF pour les enregistrements d'autorité de gestion sont disponibles pour les non-administrateurs auxquels sont accordés des droits dsp et chg sur le gestionnaire de files d'attente. Pour plus d'informations sur les droits de gestion à l'aide des commandes PCF, voir Programmable Command Formats.

Les administrateurs doivent disposer des droits requis pour que les commandes MQSC soient traitées par le gestionnaire de files d'attente éloignées. IBM MQ Explorer émet des commandes PCF pour effectuer des tâches d'administration. Les administrateurs n'ont pas besoin de droits supplémentaires pour utiliser IBM MQ Explorer afin d'administrer un gestionnaire de files d'attente sur le système local. Lorsque IBM MQ Explorer est utilisé pour administrer un gestionnaire de files d'attente sur un autre système, les administrateurs doivent disposer des droits requis pour que les commandes PCF soient traitées par le gestionnaire de files d'attente éloignées.

Avertissement : Depuis la IBM MQ 8.0, vous n'avez pas besoin d'être un administrateur pour utiliser la commande de contrôle **runmqsc**, qui émet des commandes IBM MQ Script (MQSC).

Lorsque **runmqsc** est utilisé en mode indirect pour envoyer des commandes MQSC à un gestionnaire de files d'attente éloignées, chaque commande MQSC est encapsulée dans une commande Escape PCF.

Pour plus d'informations sur les vérifications des droits d'accès lors du traitement des commandes PCF et MQSC, voir les rubriques suivantes:

- Pour les commandes PCF qui fonctionnent sur les gestionnaires de files d'attente, les files d'attente, les processus, les listes de noms et les objets d'informations d'authentification, voir Droits d'utilisation des objets IBM MQ. Reportez-vous à cette section pour connaître les commandes MQSC équivalentes encapsulées dans les commandes Escape PCF.
- Pour les commandes PCF qui fonctionnent sur des canaux, des initiateurs de canal, des programmes d'écoute et des clusters, voir Sécurité des canaux.
- Pour les commandes PCF qui fonctionnent sur des enregistrements de droits d'accès, voir Contrôle des droits d'accès pour les commandes PCF
- z/OS Pour les commandes MQSC traitées par le serveur de commandes sous IBM MQ for z/OS, voir Command security and command resource security on z/OS .

En outre, sur les systèmes Windows , le compte SYSTEM dispose d'un accès complet aux ressources IBM MQ .

Sur les plateformes UNIX and Linux , un ID utilisateur spécial **mqm** est également créé pour être utilisé par le produit uniquement. Il ne doit jamais être disponible pour les utilisateurs non privilégiés. Tous les objets IBM MQ appartiennent à l'ID utilisateur **mqm**.

Sur les systèmes Windows , les membres du groupe Administrateurs peuvent également administrer n'importe quel gestionnaire de files d'attente, tout comme le compte SYSTEM. Vous pouvez également créer un groupe **mqm** de domaine sur le contrôleur de domaine qui contient tous les ID utilisateur privilégiés actifs dans le domaine et l'ajouter au groupe **mqm** local. Certaines commandes, par exemple **crtmqm**, manipulent les droits sur les objets IBM MQ et ont donc besoin de droits pour utiliser ces objets (comme décrit dans les sections suivantes). Les membres du groupe **mqm** ont le droit d'utiliser tous les objets, mais sur les systèmes Windows , il se peut que les droits d'accès soient refusés si vous disposez d'un utilisateur local et d'un utilisateur authentifié par le domaine portant le même nom. Ceci est décrit dans «Principaux et groupes sous UNIX, Linux, and Windows», à la page 421.

Les versions de Windows comportant une fonction de contrôle de compte utilisateur restreignent les actions que les utilisateurs peuvent exécuter sur certaines fonctions du système d'exploitation, même s'ils sont membres du groupe des administrateurs. Si votre ID utilisateur se trouve dans le groupe

Administrateurs mais pas dans le groupe **mqm** , vous devez utiliser une invite de commande avec des droits élevés pour émettre des commandes d'administration IBM MQ telles que **crtmqm**. Sinon, l'erreur `AMQ7077: Vous n'êtes pas autorisé à effectuer l'opération` demandée est générée. Pour ouvrir une invite de commande avec des droits élevés, cliquez à l'aide du bouton droit de la souris sur l'option de menu Démarrer ou sur l'icône de l'invite de commande, puis sélectionnez **Exécuter en tant qu'administrateur**.

Vous n'avez pas besoin d'être membre du groupe **mqm** pour effectuer les actions suivantes:

- Emettez des commandes à partir d'un programme d'application qui émet des commandes PCF ou des commandes MQSC dans une commande PCF d'échappement, sauf si les commandes manipulent des initiateurs de canal. (Ces commandes sont décrites dans «Protection des définitions d'initialisateur de canal», à la page 113).
- Emettez des appels MQI à partir d'un programme d'application (sauf si vous souhaitez utiliser les liaisons Fast Path dans l'appel `MQCONNX`).
- Utilisez la commande `crtmqcvx` pour créer un fragment de code qui effectue la conversion de données sur les structures de type de données.
- Utilisez la commande `dspmq` pour afficher les gestionnaires de files d'attente.
- Utilisez la commande `dspmqtrc` pour afficher la sortie de trace formatée IBM MQ .

Une limitation de 12 caractères s'applique aux ID groupe et utilisateur.

Les plateformes UNIX and Linux limitent généralement la longueur d'un ID utilisateur à 12 caractères. AIX 5.3 a augmenté cette limite, mais IBM MQ continue d'observer une restriction de 12 caractères sur toutes les plateformes UNIX and Linux . Si vous utilisez un ID utilisateur de plus de 12 caractères, IBM MQ le remplace par la valeur `UNKNOWN` . Ne définissez pas d'ID utilisateur avec la valeur `UNKNOWN` .

ULW Gestion du groupe mqm sur UNIX, Linux, and Windows

Les utilisateurs du groupe mqm bénéficient de privilèges d'administration complets sur IBM MQ. Pour cette raison, vous ne devez pas inscrire d'applications et d'utilisateurs ordinaires dans le groupe mqm. Le groupe mqm doit contenir uniquement les comptes des administrateurs IBM MQ .

Ces tâches sont décrites dans:

- Windows Création et gestion de groupes sur Windows
- AIX Création et gestion de groupes sur AIX
- Solaris Création et gestion de groupes sur Solaris
- Linux Création et gestion de groupes sur Linux

Windows Si votre contrôleur de domaine s'exécute sous Windows 2000 ou Windows 2003 ou version ultérieure, votre administrateur de domaine devra peut-être configurer un compte spécial à utiliser par IBM MQ . Pour plus d'informations, voir Configuration de IBM MQ avec Prepare IBM MQ Wizard et Création et configuration de comptes de domaine Windows pour IBM MQ.

ULW Droits d'utiliser les objets IBM MQ sur UNIX, Linux, and Windows

Tous les objets sont protégés par IBM MQet les principaux doivent disposer des droits appropriés pour y accéder. Les différents principaux ont besoin de droits d'accès différents à des objets différents.

Les gestionnaires de files d'attente, les files d'attente, les définitions de processus, les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services et les objets d'informations d'authentification sont tous accessibles à partir d'applications qui utilisent des appels MQI ou des commandes PCF. Ces ressources sont toutes protégées par IBM MQet les applications doivent être autorisées à y accéder. L'entité qui effectue la demande peut être un utilisateur, un programme d'application qui émet un appel MQI ou un programme d'administration qui émet une commande PCF. L'identificateur du demandeur est appelé *principal*.

Différents groupes de principaux peuvent être accordés à différents types de droits d'accès sur le même objet. Par exemple, pour une file d'attente spécifique, un groupe peut être autorisé à effectuer des opérations d'insertion et d'extraction ; un autre groupe peut être autorisé uniquement à parcourir la file d'attente (MQGET avec l'option de navigation). De même, certains groupes peuvent avoir des droits d'insertion et d'obtention sur une file d'attente, mais ils ne sont pas autorisés à modifier les attributs de la file d'attente ou à la supprimer.

Certaines opérations sont particulièrement sensibles et doivent être limitées aux utilisateurs privilégiés. Exemple :

- Accès à certaines files d'attente spéciales, telles que les files d'attente de transmission ou la file d'attente de commandes SYSTEM.ADMIN.COMMAND.QUEUE
- Exécution de programmes qui utilisent des options de contexte MQI complètes
- Création et suppression de files d'attente d'application

Le droit d'accès complet à un objet est automatiquement accordé à l'ID utilisateur qui a créé l'objet et à tous les membres du groupe mqm (ainsi qu'aux membres du groupe Administrateurs locaux sur les systèmes Windows).

Concepts associés

«Droit d'administration de IBM MQ sur UNIX, Linux, and Windows», à la page 416
Les administrateurs IBM MQ peuvent utiliser toutes les commandes IBM MQ et accorder des droits à d'autres utilisateurs. Lorsque les administrateurs émettent des commandes pour les gestionnaires de files d'attente éloignées, ils doivent disposer des droits requis sur le gestionnaire de files d'attente éloignées. D'autres considérations s'appliquent aux systèmes Windows .

ULW Lorsque des contrôles de sécurité sont effectués sur UNIX, Linux, and Windows

Les contrôles de sécurité sont généralement effectués lors de la connexion à un gestionnaire de files d'attente, de l'ouverture ou de la fermeture d'objets et de l'insertion ou de l'obtention de messages.

Les contrôles de sécurité effectués pour une application standard sont les suivants:

Connexion au gestionnaire de files d'attente (appels MQCONN ou MQCONNX)
C'est la première fois que l'application est associée à un gestionnaire de files d'attente particulier. Le gestionnaire de files d'attente interroge l'environnement d'exploitation pour découvrir l'ID utilisateur associé à l'application. IBM MQ vérifie ensuite que l'ID utilisateur est autorisé à se connecter au gestionnaire de files d'attente et conserve l'ID utilisateur pour des vérifications ultérieures.

Les utilisateurs n'ont pas besoin de se connecter à IBM MQ; IBM MQ suppose que les utilisateurs se sont connectés au système d'exploitation sous-jacent et qu'ils ont été authentifiés par ce système.

Ouverture de l'objet (appels MQOPEN ou MQPUT1)
Les objets IBM MQ sont accessibles en ouvrant l'objet et en exécutant des commandes sur celui-ci. Toutes les vérifications de ressources sont effectuées lorsque l'objet est ouvert, plutôt que lorsqu'il est réellement consulté. Cela signifie que la demande **MQOPEN** doit spécifier le type d'accès requis (par exemple, si l'utilisateur souhaite uniquement parcourir l'objet ou effectuer une mise à jour comme placer des messages dans une file d'attente).

IBM MQ vérifie la ressource nommée dans la demande **MQOPEN** . Pour un alias ou un objet de file d'attente éloignée, l'autorisation utilisée est celle de l'objet lui-même, et non celle de la file d'attente dans laquelle l'alias ou la file d'attente éloignée est résolu. Cela signifie que l'utilisateur n'a pas besoin de droits pour y accéder. Limitez les droits de création de files d'attente aux utilisateurs privilégiés. Si vous ne le faites pas, les utilisateurs peuvent ignorer le contrôle d'accès normal simplement en créant un alias. Si une file d'attente éloignée est référencée explicitement avec les noms de file d'attente et de gestionnaire de files d'attente, la file d'attente de transmission associée au gestionnaire de files d'attente éloignées est vérifiée.

Les droits d'accès à une file d'attente dynamique sont basés sur ceux de la file d'attente modèle dont elle est dérivée, mais ne sont pas nécessairement les mêmes. Ceci est décrit dans la remarque «1», à la page 133.

L'ID utilisateur utilisé par le gestionnaire de files d'attente pour les contrôles d'accès est l'ID utilisateur obtenu à partir de l'environnement d'exploitation de l'application connectée au gestionnaire de files d'attente. Une application dûment autorisée peut émettre un appel **MQOPEN** en spécifiant un autre ID utilisateur ; des vérifications de contrôle d'accès sont ensuite effectuées sur l'autre ID utilisateur. Cela ne modifie pas l'ID utilisateur associé à l'application, mais uniquement celui utilisé pour les vérifications de contrôle d'accès.

Insertion et obtention de messages (appels MQPUT ou MQGET)
Aucune vérification de contrôle d'accès n'est effectuée.

Fermeture de l'objet (MQCLOSE)
Aucune vérification de contrôle d'accès n'est effectuée, sauf si **MQCLOSE** entraîne la suppression d'une file d'attente dynamique. Dans ce cas, il est vérifié que l'ID utilisateur est autorisé à supprimer la file d'attente.

Abonnement à une rubrique (MQSUB)
Lorsqu'une application s'abonne à une rubrique, elle spécifie le type d'opération qu'elle doit effectuer. Il s'agit soit de créer un nouvel abonnement, soit de modifier un abonnement existant, soit de reprendre un abonnement existant sans le modifier. Pour chaque type d'opération, le gestionnaire de files d'attente vérifie que l'ID utilisateur associé à l'application est autorisé à effectuer l'opération.

Lorsqu'une application s'abonne à une rubrique, les vérifications des droits d'accès sont effectuées sur les objets de rubrique qui se trouvent dans l'arborescence de rubriques au niveau ou au-dessus du point de l'arborescence de rubriques auquel l'application s'est abonnée. Les vérifications des droits d'accès peuvent impliquer des vérifications sur plusieurs objets de rubrique.

L'ID utilisateur utilisé par le gestionnaire de files d'attente pour les vérifications des droits d'accès est l'ID utilisateur obtenu auprès du système d'exploitation lorsque l'application se connecte au gestionnaire de files d'attente.

Le gestionnaire de files d'attente effectue des vérifications des droits d'accès sur les files d'attente d'abonné, mais pas sur les files d'attente gérées.

ULW Comment le contrôle d'accès est implémenté par IBM MQ sur UNIX, Linux, and Windows

IBM MQ utilise les services de sécurité fournis par le système d'exploitation sous-jacent, à l'aide du gestionnaire des droits d'accès aux objets. IBM MQ fournit des commandes pour créer et gérer des listes de contrôle d'accès.

Une interface de contrôle d'accès appelée Interface de service d'autorisation fait partie de IBM MQ. IBM MQ fournit une implémentation d'un gestionnaire de contrôle d'accès (conforme à l'interface de service d'autorisation) appelé *gestionnaire des droits d'accès aux objets (OAM)*. Elle est automatiquement installée et activée pour chaque gestionnaire de files d'attente que vous créez, sauf indication contraire (comme décrit dans «Prévention des contrôles d'accès de sécurité sur les systèmes UNIX, Linux, and Windows», à la page 373). La méthode d'accès aux objets (OAM) peut être remplacée par tout composant écrit par un utilisateur ou un fournisseur conforme à l'interface de service d'autorisation.

La méthode d'accès aux objets (OAM) exploite les fonctions de sécurité du système d'exploitation sous-jacent, à l'aide des ID utilisateur et de groupe du système d'exploitation. Les utilisateurs ne peuvent accéder aux objets IBM MQ que s'ils disposent des droits appropriés. «Contrôle de l'accès aux objets à l'aide de la méthode d'accès aux objets (OAM) sous UNIX, Linux, and Windows», à la page 362 décrit comment accorder et révoquer ces droits.

La méthode d'accès aux objets (OAM) gère une liste de contrôle d'accès (ACL) pour chaque ressource qu'elle contrôle. Les données d'autorisation sont stockées dans une file d'attente locale appelée SYSTEM.AUTH.DATA.QUEUE. L'accès à cette file d'attente est limité aux utilisateurs du groupe mqm, ainsi qu'à Windows, aux utilisateurs du groupe Administrateurs et aux utilisateurs connectés avec l'ID SYSTEM. L'accès utilisateur à la file d'attente ne peut pas être modifié.

IBM MQ fournit des commandes pour créer et gérer des listes de contrôle d'accès. Pour plus d'informations sur ces commandes, voir «Contrôle de l'accès aux objets à l'aide de la méthode d'accès aux objets (OAM) sous UNIX, Linux, and Windows», à la page 362.

IBM MQ transmet à la méthode d'accès aux objets (OAM) une demande contenant un principal, un nom de ressource et un type d'accès. La méthode d'accès aux objets (OAM) accorde ou rejette l'accès en fonction de la liste de contrôle d'accès qu'elle gère. IBM MQ suit la décision de la méthode d'accès aux objets (OAM) ; si la méthode d'accès aux objets (OAM) ne peut pas prendre de décision, IBM MQ n'autorise pas l'accès.

ULW Identification de l'ID utilisateur sous UNIX, Linux, and Windows

Le gestionnaire des droits d'accès aux objets identifie le principal qui demande l'accès à une ressource. L'ID utilisateur utilisé comme principal varie en fonction du contexte.

Le gestionnaire des droits d'accès aux objets (OAM) doit pouvoir identifier qui demande l'accès à une ressource particulière. IBM MQ utilise le terme *principal* pour désigner cet identificateur. Le principal est établi lorsque l'application se connecte pour la première fois au gestionnaire de files d'attente ; il est déterminé par le gestionnaire de files d'attente à partir de l'ID utilisateur associé à l'application de connexion. (Si l'application émet des appels XA sans se connecter au gestionnaire de files d'attente, l'ID utilisateur associé à l'application qui émet l'appel xa_open est utilisé pour les vérifications des droits d'accès par le gestionnaire de files d'attente.)

Sur les systèmes UNIX and Linux , les routines d'autorisation vérifient l'ID utilisateur réel (loggedin) ou l'ID utilisateur effectif associé à l'application. L'ID utilisateur vérifié peut dépendre du type de liaison. Pour plus de détails, voir Services optionnels.

IBM MQ propage l'ID utilisateur reçu du système dans l'en-tête de message (structure MQMD) de chaque message en tant qu'identification de l'utilisateur. Cet identificateur fait partie des informations de contexte de message et est décrit dans «Droits de contexte sur UNIX, Linux, and Windows», à la page 423. Les applications ne peuvent pas modifier ces informations sauf si elles ont été autorisées à modifier les informations de contexte.

ULW *Principaux et groupes sous UNIX, Linux, and Windows*

Les principaux peuvent appartenir à des groupes. En accordant l'accès aux ressources à des groupes plutôt qu'à des individus, vous pouvez réduire la quantité d'administration requise. Les listes de contrôle d'accès (ACL) sont basées à la fois sur les groupes et les ID utilisateur.

Par exemple, vous pouvez définir un groupe composé d'utilisateurs qui souhaitent exécuter une application particulière. Les autres utilisateurs peuvent avoir accès à toutes les ressources dont ils ont besoin en ajoutant leur ID utilisateur au groupe approprié.

Ce processus de définition et de gestion de groupes est décrit pour des plateformes particulières:

- Windows Création et gestion de groupes sur Windows
- AIX Création et gestion de groupes sur AIX
- Solaris Création et gestion de groupes sur Solaris
- Linux Création et gestion de groupes sur Linux

Un principal peut appartenir à plusieurs groupes (son ensemble de groupes). Il dispose de l'ensemble des droits accordés à chaque groupe de son groupe. Ces droits étant mis en cache, les modifications apportées à l'appartenance au groupe du principal ne sont pas reconnues tant que le gestionnaire de files d'attente n'est pas redémarré, sauf si vous émettez la commande MQSC **REFRESH SECURITY** (ou son équivalent PCF).

Linux UNIX Systèmes UNIX and Linux

Depuis la IBM MQ 8.0, les listes de contrôle d'accès (ACL) sont basées sur les ID utilisateur et les groupes et vous pouvez utiliser l'un ou l'autre pour l'autorisation en définissant l'attribut

SecurityPolicy sur la valeur appropriée, comme décrit dans Configuration des services installables et Configuration des sections de service d'autorisation sous UNIX et Linux.

Depuis la IBM MQ 8.0, vous pouvez utiliser le *modèle basé sur l'utilisateur* pour l'autorisation, ce qui vous permet d'utiliser à la fois des utilisateurs et des groupes. Toutefois, lorsque vous spécifiez un utilisateur dans la commande setmqaut , les nouveaux droits s'appliquent à cet utilisateur seul et non aux groupes auxquels cet utilisateur appartient. Pour plus d'informations, voir OAM user-based permissions on UNIX and Linux systems.

Lorsque vous utilisez le *modèle basé sur un groupe* pour l'autorisation, le groupe principal auquel appartient l'ID utilisateur est inclus dans la liste de contrôle d'accès. L'ID utilisateur individuel n'est pas inclus et des droits sont accordés à tous les membres de ce groupe. Pour cette raison, sachez que vous pouvez modifier par inadvertance les droits d'un principal en modifiant les droits d'un autre principal du même groupe.

Tous les utilisateurs sont nominalement affectés au groupe d'utilisateurs par défaut `personne` et, par défaut, aucune autorisation n'est accordée à ce groupe. Vous pouvez modifier l'autorisation dans le groupe `personne` pour accorder l'accès aux ressources IBM MQ à des utilisateurs sans autorisation spécifique.

Ne définissez pas d'ID utilisateur avec la valeur `UNKNOWN`. La valeur `UNKNOWN` est utilisée lorsqu'un ID utilisateur est trop long. Par conséquent, les ID utilisateur arbitraires utilisent les droits d'accès de `UNKNOWN`.

Les ID utilisateur peuvent contenir jusqu'à 12 caractères et les noms de groupe jusqu'à 12 caractères.

Windows Systèmes Windows

Les listes de contrôle d'accès sont basées sur les ID utilisateur et les groupes. Les vérifications sont les mêmes que pour UNIX. Vous pouvez avoir différents utilisateurs sur différents domaines avec le même ID utilisateur. IBM MQ permet aux ID utilisateur d'être qualifiés par un nom de domaine afin que ces utilisateurs puissent disposer de différents niveaux d'accès.

Le nom de groupe peut éventuellement inclure un nom de domaine, spécifié dans les formats suivants:

```
GroupName@domain domain_name\group_name
```

Les groupes globaux sont vérifiés par la méthode d'accès aux objets (OAM) dans deux cas uniquement:

1. La section de sécurité du gestionnaire de files d'attente inclut le paramètre: `GroupModel=GlobalGroups`. Voir Sécurisation.
2. Le gestionnaire de files d'attente utilise un autre groupe d'accès de sécurité. Voir **crtmqm** .

Les ID utilisateur peuvent contenir jusqu'à 20 caractères, les noms de domaine jusqu'à 15 caractères et les noms de groupe jusqu'à 64 caractères.

La méthode d'accès aux objets (OAM) vérifie d'abord la base de données de sécurité locale, puis la base de données du domaine principal, et enfin la base de données des domaines de confiance. Le premier ID utilisateur rencontré est utilisé par la méthode d'accès aux objets (OAM) pour la vérification. Chacun de ces ID utilisateur peut avoir des appartenances de groupe différentes sur un ordinateur particulier.

Certaines commandes de contrôle (par exemple, **crtmqm**) modifient les droits sur les objets IBM MQ à l'aide du gestionnaire des droits d'accès aux objets (OAM). La méthode d'accès aux objets (OAM) recherche les bases de données de sécurité dans l'ordre indiqué dans le paragraphe précédent afin de déterminer les droits d'accès pour un ID utilisateur particulier. Par conséquent, les droits d'accès déterminés par la méthode d'accès aux objets (OAM) peuvent remplacer le fait qu'un ID utilisateur soit membre du groupe mqm local. Par exemple, si vous émettez la commande **crtmqm** à partir d'un ID utilisateur authentifié par un contrôleur de domaine qui est membre du groupe mqm local via un groupe global, la commande échoue si le système possède un utilisateur local du même nom qui ne fait pas partie du groupe mqm local.

Pour plus d'informations sur la définition de l'attribut **SecurityPolicy** sous Windows, voir Services installables et Configuration des sections de service d'autorisation sous Windows.

Windows *Identificateurs de sécurité (SID) Windows*

IBM MQ on Windows utilise le SID où il est disponible. Si un SID Windows n'est pas fourni avec une demande d'autorisation, IBM MQ identifie l'utilisateur en fonction du nom d'utilisateur uniquement, mais cela peut entraîner l'octroi de droits d'accès incorrects.

Sur les systèmes Windows , l'identificateur de sécurité (SID) est utilisé pour compléter l'ID utilisateur. Le SID contient des informations qui identifient les détails complets du compte utilisateur sur la base de données du gestionnaire de compte de sécurité Windows (SAM) dans laquelle l'utilisateur est défini. Lorsqu'un message est créé sur IBM MQ for Windows, IBM MQ stocke le SID dans le descripteur de message. Lorsque IBM MQ on Windows effectue des vérifications d'autorisation, il utilise le SID pour interroger les informations complètes de la base de données SAM. (La base de données SAM dans laquelle l'utilisateur est défini doit être accessible pour que cette requête aboutisse.)

Par défaut, si un SID Windows n'est pas fourni avec une demande d'autorisation, IBM MQ identifie l'utilisateur en se basant uniquement sur le nom d'utilisateur. Pour ce faire, il effectue des recherches dans les bases de données de sécurité dans l'ordre suivant:

1. Base de données de sécurité locale
2. Base de données de sécurité du domaine principal
3. Base de données de sécurité des domaines de confiance

Si le nom d'utilisateur n'est pas unique, des droits IBM MQ incorrects peuvent être accordés. Pour éviter ce problème, incluez un SID dans chaque demande d'autorisation ; le SID est utilisé par IBM MQ pour établir les données d'identification de l'utilisateur.

Pour indiquer que toutes les demandes d'autorisation doivent inclure un SID, utilisez **regedit**. Définissez SecurityPolicy sur NTSIDsRequired.

ULW Droits d'utilisateur de remplacement sous UNIX, Linux, and Windows

Vous pouvez indiquer qu'un ID utilisateur peut utiliser les droits d'un autre utilisateur lors de l'accès à un objet IBM MQ . Il s'agit des *droits d'accès utilisateur de remplacement*, que vous pouvez utiliser sur n'importe quel objet IBM MQ .

Les droits d'utilisateur de remplacement sont essentiels lorsqu'un serveur reçoit des demandes d'un programme et souhaite s'assurer que le programme dispose des droits requis pour la demande. Le serveur peut disposer des droits requis, mais il doit savoir si le programme dispose des droits requis pour les actions qu'il a demandées.

Par exemple, supposons qu'un programme serveur s'exécutant sous l'ID utilisateur PAYSERV extrait un message de demande d'une file d'attente qui a été placée dans la file d'attente par l'ID utilisateur USER1. Lorsque le programme serveur obtient le message de demande, il traite la demande et insère la réponse dans la file d'attente de réponse spécifiée dans le message de demande. Au lieu d'utiliser son propre ID utilisateur (PAYSERV) pour autoriser l'ouverture de la file d'attente de réponse, le serveur peut spécifier un ID utilisateur différent, dans ce cas, USER1. Dans cet exemple, vous pouvez utiliser les droits d'utilisateur de remplacement pour contrôler si PAYSERV est autorisé à spécifier USER1 comme ID utilisateur de remplacement lorsqu'il ouvre la file d'attente de réponse.

L'ID utilisateur de remplacement est indiqué dans la zone **AlternateUserId** du descripteur d'objet.

ULW Droits de contexte sur UNIX, Linux, and Windows

Le contexte est une information qui s'applique à un message particulier et qui est contenue dans le descripteur de message, MQMD, qui fait partie du message. Les applications peuvent spécifier les données contextuelles lorsqu'un appel MQOPEN ou MQPUT est effectué.

Les informations contextuelles comportent deux sections :

La section d'identité
D'où vient le message. Il se compose des zones `UserIdentifier`, `AccountingTokenet` `ApplIdentityData` .

Section d'origine
D'où provient le message et quand il a été placé dans la file d'attente. Il se compose des zones `PutApplType`, `PutApplName`, `PutDate`, `PutTimeet` `ApplOriginData` .

Les applications peuvent spécifier les données contextuelles lorsqu'un appel `MQOPEN` ou `MQPUT` est effectué. Ces données peuvent être générées par l'application, transmises à partir d'un autre message ou générées par le gestionnaire de files d'attente par défaut. Par exemple, les données contextuelles peuvent être utilisées par les programmes serveur pour vérifier l'identité du demandeur, en vérifiant si le message provient d'une application s'exécutant sous un ID utilisateur autorisé.

Un programme serveur peut utiliser le `UserIdentifier` pour déterminer l'ID utilisateur d'un autre utilisateur. Vous utilisez l'autorisation de contexte pour contrôler si l'utilisateur peut spécifier l'une des options de contexte dans un appel `MQOPEN` ou `MQPUT1` .

Voir Contrôle des informations de contexte pour plus d'informations sur les options de contexte et Présentation de MQMD pour obtenir des descriptions des zones de descripteur de message relatives au contexte.

Implémentation du contrôle d'accès dans les exits de sécurité

Vous pouvez implémenter le contrôle d'accès dans un exit de sécurité à l'aide de MCAUserIdentifier ou du gestionnaire des droits d'accès aux objets.

MCAUserIdentifier

Chaque instance d'un canal en cours est associée à une structure de définition de canal, MQCD. Les valeurs initiales des zones de MQCD sont déterminées par la définition de canal créée par un administrateur IBM MQ . En particulier, la valeur initiale de l'une des zones, *MCAUserIdentifier*, est déterminée par la valeur du paramètre MCAUSER dans la commande DEFINE CHANNEL ou par l'équivalent de MCAUSER si la définition de canal est créée d'une autre manière.

La structure MQCD est transmise à un programme d'exit de canal lorsqu'elle est appelée par un agent MCA. Lorsqu'un exit de sécurité est appelé par un agent MCA, l'exit de sécurité peut modifier la valeur de *MCAUserIdentifier*, en remplaçant toute valeur spécifiée dans la définition de canal.

Multi Sous Multiplateformes, sauf si la valeur de *MCAUserIdentifier* est vide, le gestionnaire de files d'attente utilise la valeur de *MCAUserIdentifier* comme ID utilisateur pour les vérifications des droits d'accès lorsqu'un agent MCA tente d'accéder aux ressources du gestionnaire de files d'attente après s'être connecté au gestionnaire de files d'attente. Si la valeur de *MCAUserIdentifier* est vide, le gestionnaire de files d'attente utilise à la place l'ID utilisateur par défaut de l'agent MCA. Cela s'applique aux canaux RCVR, RQSTR, CLUSRCVR et SVRCONN. Pour l'envoi d'agents MCA, l'ID utilisateur par défaut est toujours utilisé pour les vérifications des droits d'accès, même si la valeur de *MCAUserIdentifier* n'est pas vide.

z/OS Sous z/OS, le gestionnaire de files d'attente peut utiliser la valeur de *MCAUserIdentifier* pour les vérifications des droits d'accès, à condition qu'elle ne soit pas vide. Pour la réception des MCM et des MCM de connexion au serveur, le fait que le gestionnaire de files d'attente utilise ou non la valeur de *MCAUserIdentifier* pour les vérifications des droits d'accès dépend des éléments suivants:

- Valeur du paramètre PUTAUT dans la définition de canal
- Profil RACF utilisé pour les vérifications
- Niveau d'accès de l'ID utilisateur de l'espace adresse de l'initiateur de canal au profil RESLEVEL

Pour l'envoi des MCM, il dépend des éléments suivants:

- Indique si l'agent MCA émetteur est un appelant ou un répondeur
- Niveau d'accès de l'ID utilisateur de l'espace adresse de l'initiateur de canal au profil RESLEVEL

L'ID utilisateur stocké par un exit de sécurité dans *MCAUserIdentifier* peut être acquis de différentes manières. Voici quelques exemples :

- A condition qu'il n'y ait pas d'exit de sécurité à l'extrémité client d'un canal MQI, un ID utilisateur associé à l'application client IBM MQ est transmis de l'agent MCA de connexion client à l'agent MCA de connexion serveur lorsque l'application client émet un appel MQCONN.L'agent MCA de connexion serveur stocke cet ID utilisateur dans la zone *RemoteUserIdentifier* de la structure de définition de canal, MQCD. Si la valeur de *MCAUserIdentifier* est vide à ce stade, l'agent MCA stocke le même ID utilisateur dans *MCAUserIdentifier*. Si l'agent MCA ne stocke pas l'ID utilisateur dans *MCAUserIdentifier*, un exit de sécurité peut le faire ultérieurement en affectant à *MCAUserIdentifier* la valeur *RemoteUserIdentifier*.

  Si l'ID utilisateur qui provient du système client entre dans un nouveau domaine de sécurité et n'est pas valide sur le système serveur, l'exit de sécurité peut remplacer l'ID utilisateur par un ID utilisateur valide et stocker l'ID utilisateur remplacé dans *MCAUserIdentifier*.
- L'ID utilisateur peut être envoyé par l'exit de sécurité partenaire dans un message de sécurité.

  Sur un canal de transmission de messages, un exit de sécurité appelé par l'agent MCA émetteur peut envoyer l'ID utilisateur sous lequel l'agent MCA émetteur s'exécute. Un exit de sécurité appelé par l'agent MCA récepteur peut ensuite stocker l'ID utilisateur dans *MCAUserIdentifier*. De même, sur un canal MQI, un exit de sécurité à l'extrémité client du canal peut envoyer l'ID utilisateur associé à l'application IBM MQ MQI client . Un exit de sécurité à l'extrémité serveur du canal peut ensuite stocker l'ID utilisateur dans *MCAUserIdentifier*. Comme dans l'exemple précédent, si l'ID utilisateur n'est pas valide sur le système cible, l'exit de sécurité peut remplacer l'ID utilisateur par un ID utilisateur valide et stocker l'ID utilisateur remplacé dans *MCAUserIdentifier*.

  Si un certificat numérique est reçu dans le cadre du service d'identification et d'authentification, un exit de sécurité peut mapper le nom distinctif du certificat à un ID utilisateur valide sur le système cible. Il peut ensuite stocker l'ID utilisateur dans *MCAUserIdentifier*.
- Si TLS est utilisé sur le canal, le nom distinctif (DN) du partenaire est transmis à l'exit dans la zone Ptr SSLPeerNamede MQCD, et le nom distinctif de l'émetteur de ce certificat est transmis à l'exit dans la zone Ptr SSLRemCertIssNamede MQCXP.

Pour plus d'informations sur la zone *MCAUserIdentifier* , la structure de définition de canal, MQCD et la structure de paramètre d'exit de canal, MQCXP, voir Appels d'exit de canal et structures de données. Pour plus d'informations sur l'ID utilisateur qui provient d'un système client sur un canal MQI, voir Contrôle d'accès.

Remarque : Les applications d'exit de sécurité construites avant l'édition de IBM WebSphere MQ 7.1 peuvent nécessiter une mise à jour. Pour plus d'informations, voir Programmes d'exit de sécurité de canal.

Authentification d'utilisateur du gestionnaire des droits d'accès aux objets IBM MQ

Sur les connexions IBM MQ MQI client , les exits de sécurité peuvent être utilisés pour modifier ou créer la structure MQCSP utilisée dans l'authentification d'utilisateur du gestionnaire des droits d'accès aux objets (OAM). Ceci est décrit dans Programmes d'exit de canal pour les canaux de messagerie

Implémentation du contrôle d'accès dans les exits de message

Vous devrez peut-être utiliser un exit de message pour remplacer un ID utilisateur par un autre.

Prenons l'exemple d'une application client qui envoie un message à une application serveur. L'application serveur peut extraire l'ID utilisateur de la zone *UserIdentifier* du descripteur de message et, à condition qu'elle dispose de droits d'utilisateur de remplacement, demander au gestionnaire de files d'attente d'utiliser cet ID utilisateur pour les vérifications des droits d'accès lorsqu'il accède aux ressources IBM MQ pour le compte du client.

Si le paramètre PUTAUT est défini sur CTX (ou ALTMCA sous z/OS) dans la définition de canal, l'ID utilisateur dans la zone *UserIdentifier* de chaque message entrant est utilisé pour les vérifications des droits d'accès lorsque l'agent MCA ouvre la file d'attente de destination.

Dans certains cas, lorsqu'un message de rapport est généré, il est placé à l'aide des droits de l'ID utilisateur dans la zone *UserIdentifier* du message à l'origine du rapport. En particulier, les rapports de confirmation à la livraison (COD) et les rapports d'expiration sont toujours soumis à cette autorité.

En raison de ces situations, il peut être nécessaire de remplacer un ID utilisateur par un autre dans la zone *UserIdentifier* lorsqu'un message entre dans un nouveau domaine de sécurité. Ceci peut être réalisé par un exit de message à l'extrémité réceptrice du canal. Vous pouvez également vous assurer que l'ID utilisateur dans la zone *UserIdentifier* d'un message entrant est défini dans le nouveau domaine de sécurité.

Si un message entrant contient un certificat numérique pour l'utilisateur de l'application qui a envoyé le message, un exit de message peut valider le certificat et mapper le nom distinctif du certificat à un ID utilisateur valide sur le système de réception. Il peut ensuite définir la zone *UserIdentifier* du descripteur de message sur cet ID utilisateur.

S'il est nécessaire qu'un exit de message modifie la valeur de la zone *UserIdentifier* dans un message entrant, il peut être approprié que l'exit de message authentifie l'expéditeur du message en même temps. Pour plus de détails, voir «Mappage d'identité dans les exits de message», à la page 346.

Implémentation du contrôle d'accès dans l'exit d'API et l'exit de croisement d'API

Une API ou un exit de croisement d'API peut fournir des contrôles d'accès pour compléter ceux fournis par IBM MQ. En particulier, l'exit peut fournir un contrôle d'accès au niveau du message. L'exit peut s'assurer qu'une application insère dans une file d'attente, ou extrait d'une file d'attente, uniquement les messages qui répondent à certains critères.

Prenons les exemples suivants :

- Un message contient des informations sur une commande. Lorsqu'une application tente d'insérer un message dans une file d'attente, une API ou un exit de croisement d'API peut vérifier que la valeur totale de la commande est inférieure à une limite prescrite.
- Les messages arrivent dans une file d'attente de destination à partir de gestionnaires de files d'attente éloignées. Lorsqu'une application tente d'extraire un message de la file d'attente, une API ou un exit de croisement d'API peut vérifier que l'expéditeur du message est autorisé à envoyer un message à la file d'attente.

Autorisation LDAP

Vous pouvez utiliser l'autorisation LDAP pour supprimer la nécessité d'un ID utilisateur local.

Disponibilité de l'autorisation LDAP sur les plateformes prises en charge

L'autorisation LDAP est disponible sur les plateformes suivantes :

- UNIX
- IBM i
- Windows

Avertissement :

A partir de la disponibilité générale d' IBM MQ 9.0 , cette fonctionnalité est disponible sur tous les gestionnaires de files d'attente, qu'ils soient nouveaux ou migrés à partir d'une édition antérieure.

Présentation de l'autorisation LDAP

Avec l'autorisation LDAP, les commandes qui gèrent la configuration d'autorisation, telles que **setmqaut** et **DISPLAY AUTHREC**, peuvent traiter les noms distinctifs. Auparavant, les utilisateurs étaient

authentifiés en comparant leurs données d'identification avec le nombre maximal de caractères disponibles pour les utilisateurs et les groupes sur le système d'exploitation local.

Avertissement : Si vous avez exécuté la commande **DEFINE AUTHINFO** , vous devez redémarrer le gestionnaire de files d'attente. Si vous ne redémarrez pas le gestionnaire de files d'attente, la commande setmqaut ne renvoie pas le résultat correct.

Si un utilisateur fournit un ID utilisateur plutôt qu'un nom distinctif, l'ID utilisateur est traité. Par exemple, lorsqu'il existe un message entrant sur un canal avec PUTAUT (CTX), les caractères de l'ID utilisateur sont mappés à un nom distinctif LDAP et les vérifications d'autorisation appropriées sont effectuées.

D'autres commandes, telles que **DISPLAY CONN**, continuent d'utiliser et d'afficher la valeur réelle de l'ID utilisateur, même si cet ID utilisateur n'existe pas sur le système d'exploitation local.

UNIX Lorsque l'autorisation LDAP est en place, le gestionnaire de files d'attente utilise toujours le modèle utilisateur de sécurité sur les plateformes UNIX , quel que soit l'attribut **SecurityPolicy** dans le fichier `qm.ini` . Par conséquent, la définition des droits d'accès d'un utilisateur individuel n'affecte que cet utilisateur, et personne d'autre n'appartient à l'un des groupes de cet utilisateur.

Comme pour le modèle de système d'exploitation, un utilisateur dispose toujours des droits combinés qui ont été affectés à la fois à l'individu et à tous les groupes (le cas échéant) auxquels l'utilisateur appartient.

Par exemple, supposons que les enregistrements suivants ont été définis dans un référentiel LDAP.

- Dans la classe **inetOrgPerson** :

```
dn="cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
        email=JohnDoe1@yourcompany.com [longer than 12 characters]
        shortu=jodoe
        Phone=1234567
```

- Dans la classe **groupOfNames** :

```
dn="cn=Application Group A, ou=groups, o=yourcompany, c=yourcountry"
        longname=ApplicationGroupA [longer than 12 characters]
        members="cn=JaneDoe, ou=users, o=yourcompany, c=yourcountry",
                "cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
```

A des fins d'authentification, un gestionnaire de files d'attente utilisant ce serveur LDAP doit avoir été défini de sorte que sa valeur **CONNAUTH** pointe vers un objet **AUTHINFO** de type IDPWLDAP, et dont les attributs de résolution de nom appropriés sont probablement définis comme suit:

```
USRFIELD(email) SHORTUSR(shortu)
BASEDNU(ou=users,o=yourcompany,c=yourcountry) CLASSUSR(inetOrgPerson)
```

Etant donné cette configuration pour l'authentification, une application peut renseigner la zone CSPUserID , utilisée dans l'appel MQCNO, avec l'un des ensembles de valeurs suivants:

```
" cn=JohnDoe ", " JohnDoe1@yourcompany.com ", " email=JohnDoe1@yourcompany.com "
```

ou

```
" cn=JohnDoe, ou=users, o=ibm, c=uk ", " shortu=jodoe "
```

Dans les deux cas, le système peut utiliser les valeurs fournies pour authentifier le contexte de système d'exploitation de " jodoe".

Définition des autorisations

Comment utiliser le nom abrégé ou **USRFIELD** pour définir les autorisations.

L'approche de l'utilisation de plusieurs formats, décrite dans «Autorisation LDAP», à la page 426, se poursuit avec les commandes d'autorisation, avec une extension supplémentaire que `shortname` ou USRFIELD peut être utilisé de manière non décorée.

La chaîne de caractères spécifie un attribut particulier dans l'enregistrement LDAP lors de la désignation des utilisateurs (principaux) pour l'autorisation.

Important : La chaîne de caractères ne doit pas contenir le caractère = , car ce caractère ne peut pas être utilisé dans un ID utilisateur du système d'exploitation.

Si vous transmettez un nom de principal à la méthode d'accès aux objets (OAM) pour une autorisation qui peut être `shortname`, la chaîne de caractères doit contenir 12 caractères. L'algorithme de mappage tente d'abord de le résoudre en nom distinctif à l'aide de l'attribut SHORTUSR dans sa requête LDAP.

Si cela échoue avec une erreur UNKNOWN_ENTITY ou si la chaîne donnée ne peut pas être un `shortname`, une nouvelle tentative est effectuée à l'aide de l'attribut USRFIELD pour construire la requête LDAP.

Avertissement : Si vous avez exécuté la commande DEFINE AUTHINFO, vous devez redémarrer le gestionnaire de files d'attente. Si vous ne redémarrez pas le gestionnaire de files d'attente, la commande setmqaut ne renvoie pas le résultat correct.

Pour le traitement des autorisations utilisateur, les paramètres de commande setmqaut suivants sont tous équivalents.

Tableau 72. Paramètres d'autorisation de l'utilisateur

Commande	Remarque
`setmqaut -m QM -t qmgr -p jodoe +connect`	Il s'agit d'un nom non qualifié non hiérarchique, résolu via SHORTUSR.
`setmqaut -m QM -t qmgr -pJohnDoe1@yourcompany.com +connect`	Il s'agit également d'un nom non qualifié non hiérarchique, qui se résout via USRFIELD en une même entité.
`setmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com +connect`	Utilisation d'un attribut nommé.
`setmqaut -m QM -t qmgr -p "phone=1234567" +connect`	Utilisation d'un autre attribut nommé qui ne doit pas nécessairement être l'un de ceux configurés sur l'objet AUTHINFO.

Vous pouvez utiliser la commande MQSC SET AUTHREC comme alternative à la commande **setmqaut** :

```
SET AUTHREC OBJTYPE(QMGR) PRINCIPAL('JohnDoe1@yourcompany.com') AUTHADD(connect)
```

ou la commande PCF Set Authority Record (MQCMD_SET_AUTH_REC) avec l'élément MQCACF_PRINCIPAL_ENTITY_NAMES contenant la chaîne:

```
 "cn=JohnDoe,ou=users,o=yourcompany,c=yourcountry"
```

Lors du traitement des groupes, il n'y a pas d'ambiguïté sur le traitement de `shortname` , car il n'est pas nécessaire d'insérer une forme de nom de groupe en 12 caractères. Par conséquent, il n'existe pas d'équivalent de l'attribut SHORTUSR pour les groupes.

Cela signifie que les exemples de syntaxe décrits dans Tableau 73, à la page 429 sont valides, en supposant que vous avez configuré l'objet AUTHINFO avec les attributs étendus et défini sur:

```
GRPFIELD(longname)
BASEDNG(ou=groups,o=yourcompany,c=yourcountry ) CLASSGRP(groupOfNames)
```

Tableau 73. Paramètres d'autorisation de groupe	
Commande	**Remarque**
`setmqaut -m QM -t qmgr -g ApplicationGroupA +connect`	Utilisation de GRPFIELD pour la résolution
`setmqaut -m QM -t qmgr -g longname=ApplicationGroupA +connect`	Attribution d'un nom à un attribut unique
`setmqaut -m QM -t qmgr -g "cn=Application Group A,ou=groups,o=yourcompany,c=yourcountry" +connect`	Utilisation du nom distinctif complet

Vous pouvez utiliser la commande MQSC SET AUTHREC comme alternative à la commande **setmqaut** précédente:

```
SET AUTHREC OBJTYPE(QMGR) GROUP('ApplicationGroupA')
      AUTHADD(connect)
```

ou la commande PCF Set Authority Record (MQCMD_SET_AUTH_REC) avec l'élément MQCACF_GROUP_ENTITY_NAMES contenant la chaîne:

```
 "ApplicationGroupA"
```

Important :

Quel que soit le format utilisé pour faire référence à un nom, que ce soit pour un utilisateur ou un groupe, il doit être possible de dériver un nom distinctif unique.

Par exemple, vous ne devez pas avoir deux enregistrements distincts ayant tous deux "`shortu=jodoe`".

Si un seul nom distinctif unique ne peut pas être déterminé, la méthode d'accès aux objets (OAM) renvoie MQRC_UNKNOWN_ENTITY.

Affichage des autorisations

Diverses méthodes d'affichage de l'autorisation des utilisateurs ou des groupes.

Commande dspmqaut

La méthode la plus simple pour afficher les autorisations disponibles pour un utilisateur ou un groupe consiste à utiliser la commande dspmqaut .

Vous pouvez utiliser une requête sur l'une des variantes de syntaxe pour identifier un utilisateur ou un groupe. Notez que la sortie de la commande répète l'identité dans le format indiqué sur la ligne de commande. La sortie ne signale pas le nom distinctif résolu complet.

Exemple :

```
dspmqaut -m QM -t qmgr -p johndoe
Entity johndoe has the following authorizations for object QM:
    connect
```

ou

```
dspmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com
Entity email=JohnDoe1@yourcompany.com has the following authorizations for object QM:
    connect
```

commandes dmpmqaut et dmpmqcfg

La commande dmpmqaut et ses équivalents MQSC ou PCF peuvent spécifier le principal ou le groupe dans n'importe lequel des formats pris en charge, comme les tables **setmqaut** décrites dans «Définition des autorisations», à la page 427. Cependant, contrairement à **dspmqaut**, la commande **dmpmqaut** signale toujours le nom distinctif complet.

```
dmpmqaut -m QM -t qmgr -p jodoe
-----------------------------------
profile: self
object type:qmgr
entity:cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry
entity type: principal
authority: connect
```

De même, la commande dmpmqcfg , qui ne comporte aucun filtrage sur les enregistrements sélectionnés, affiche toujours le nom distinctif complet dans un format qui peut être réexécuté ultérieurement.

```
dmpmqcfg -m QM -x authrec
-----------------------------------
SET AUTHREC PROFILE(SELF) +
    PRINCIPAL('cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry') +
    OBJTYPE(QMGR)
    AUTHADD(CONNECT)
```

Autres considérations lors de l'utilisation de l'autorisation LDAP

Brève description des modifications apportées à l'interface MQI (Message Queue Interface) et aux autres commandes MQSC et PCF dont vous devez tenir compte lors de l'utilisation de l'autorisation LDAP de IBM MQ 9.0.0.

ADOPTCTX

Il n'est pas nécessaire que les applications fournissent des informations d'authentification ou que l'attribut ADOPTCTX soit défini sur YES.

Si une application ne s'authentifie pas explicitement ou si **ADOPTCTX** est défini sur NO pour l'objet CONNAUTH actif, le contexte d'identité associé à l'application est extrait de l'ID utilisateur du système d'exploitation.

Lorsque des autorisations doivent être appliquées, ce contexte est mappé à une identité LDAP à l'aide des mêmes règles que pour les commandes setmqaut .

Paramètres d'entrée des appels MQI

MQOPEN, MQPUT1et MQSUB possèdent des structures qui permettent d'indiquer un autre ID utilisateur.

Si ces zones sont utilisées, l'ID utilisateur à 12 caractères est mappé à un nom distinctif à l'aide des mêmes règles que sur les commandes **setmqaut**, **dmpmqaut**et **dspmqaut** .

MQPUT et MQPUT1 permettent également aux programmes disposant des droits appropriés de définir la zone MQMD UserIdentifier . La valeur de cette zone n'est pas contrôlée lors du processus PUT et peut être définie sur n'importe quelle valeur.

Toutefois, comme d'habitude, la valeur **UserIdentifier** peut être utilisée pour l'autorisation à des étapes ultérieures du traitement des messages, par exemple lorsque PUTAUT (CTX) est défini sur un canal récepteur.

A ce stade, l'autorisation de l'identificateur sera vérifiée à l'aide de la configuration du gestionnaire de files d'attente de réception-qui peut être LDAP ou basé sur le système d'exploitation.

Paramètres de sortie des appels MQI

Chaque fois qu'un ID utilisateur est fourni à un programme dans une structure MQI, il s'agit de la version de nom abrégé à 12 caractères associée à la connexion.

Par exemple, la valeur **MQAXC.UserId** pour les exits API est le nom abrégé renvoyé par le mappage LDAP.

Autres commandes MQSC et PCF d'administration

Les commandes qui affichent les informations utilisateur dans le statut de l'objet, tel que DISPLAY CONN USERID, renvoient le nom abrégé de 12 caractères associé au contexte. Le nom distinctif complet n'est pas affiché.

Les commandes qui permettent l'assertion d'identités, telles que les règles de mappage CHLAUTH ou les valeurs MCAUSER pour les canaux, peuvent prendre des valeurs jusqu'à la longueur maximale définie pour ces attributs (actuellement 64 caractères).

La syntaxe n'est pas modifiée. Lorsque l'autorisation est requise pour cette identité, elle est mappée en interne à un nom distinctif à l'aide des mêmes règles que pour les commandes **setmqaut**, **dmpmqaut**et **dspmqaut** .

Cela signifie que la valeur MCAUSER sur une définition de canal peut ne pas s'afficher comme la même chaîne que DISPLAY CHSTATUS mais qu'elle fait référence à la même identité.

Exemple :

```
DEFINE CHL(SV1) CHLTYPE(SVRCONN) MCAUSER('cn=JohnDoe')
DEFINE CHL(SV2) CHLTYPE(SVRCONN) MCAUSER('jodoe')
DEFINE CHL(SV3) CHLTYPE(SVRCONN) MCAUSER('JohnDoe1@yourcompany.com')
```

DISPLAY CHSTATUS (*) ALL affiche la valeur SHORTUSR, *MCAUSER (jodoe)* pour toutes les connexions.

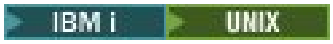
Basculement entre les modèles d'autorisation du système d'exploitation et LDAP

Comment basculer entre les différentes méthodes d'autorisation sur différentes plateformes.

L'attribut CONNAUTH du gestionnaire de files d'attente pointe vers un objet AUTHINFO. Lorsque l'objet est de type IDPWLDAP, un référentiel LDAP est utilisé pour l'authentification.

Vous pouvez maintenant appliquer une méthode d'autorisation à ce même objet, ce qui vous permet de continuer avec l'autorisation basée sur le système d'exploitation ou d'utiliser l'autorisation LDAP

Plateformes UNIX et IBM i



Le gestionnaire de files d'attente peut être permuté à tout moment entre les modèles OS et LDAP. Vous pouvez modifier la configuration et la rendre active à l'aide de la commande REFRESH SECURITY TYPE (CONNAUTH) .

Par exemple, si cet objet a déjà été configuré avec les informations de connexion pour l'authentification:

```
ALTER AUTHINFO(MYLDAP) AUTHTYPE(IDPWLDAP) +
        AUTHORMD(SEARCHGRP) +
        BASEDNG('ou=groups,o=ibm,c=uk') +
        <other attributes>
ALTER QMGR CONNAUTH(MYLDAP)
REFRESH SECURITY
```

Windows

Si une modification de la configuration des droits d'accès implique le basculement entre les modèles OS et LDAP, le gestionnaire de files d'attente doit être redémarré pour que la modification soit prise en compte. Sinon, vous pouvez activer la modification à l'aide de la commande REFRESH SECURITY TYPE (CONNAUTH) .

Règles de traitement

Lorsque vous passez de l'autorisation du système d'exploitation à l'autorisation LDAP, toutes les règles d'autorité du système d'exploitation existantes qui ont été définies deviennent inactives et invisibles.

Les commandes telles que **dmpmqaut** n'affichent pas ces règles de système d'exploitation. De même, lorsque vous revenez de LDAP au système d'exploitation, toutes les autorisations LDAP définies deviennent inactives et invisibles, ce qui restaure les règles du système d'exploitation d'origine.

Si vous souhaitez sauvegarder les définitions d'un gestionnaire de files d'attente pour une raison quelconque, à l'aide de la commande **dmpmqcfg** , cette sauvegarde contiendra uniquement les règles définies pour la méthode d'autorisation en vigueur au moment de la sauvegarde.

Administration LDAP

Présentation de la façon dont chaque plateforme administre LDAP.

Lors de l'utilisation de l'autorisation LDAP, l'appartenance au groupe `mqm` (ou équivalent) dans le système d'exploitation n'est pas si importante. Le fait d'être membre de ce groupe contrôle uniquement si certaines commandes de ligne de commande peuvent être traitées.

En particulier, vous devez faire partie de ce groupe pour exécuter les commandes strmqm et endmqm .

Une fois que le gestionnaire de files d'attente est en cours d'exécution, le compte privilégié est désormais limité. Outre l'ID utilisateur de la personne qui émet la commande **strmqm** , les autres utilisateurs appartenant au groupe `mqm` du système d'exploitation (ou équivalent) n'obtiennent pas de privilèges spéciaux.

Les autorisations des autres utilisateurs sont basées sur les groupes LDAP auxquels ils appartiennent. Une utilisation non qualifiée du nom de groupe `mqm` dans des commandes telles que **setmqaut** n'est pas autorisée à être mappée à un groupe LDAP.

UNIX

Une fois que le gestionnaire de files d'attente est en cours d'exécution, le seul compte automatiquement privilégié est l'utilisateur réel qui a démarré le gestionnaire de files d'attente.

L'ID `mqm` existe toujours et est utilisé en tant que propriétaire des ressources du système d'exploitation, telles que les fichiers, car `mqm` est l'ID effectif sous lequel le gestionnaire de files d'attente s'exécute. Toutefois, l'utilisateur `mqm` ne pourra pas effectuer automatiquement les tâches d'administration contrôlées par la méthode d'accès aux objets (OAM).

IBM i

Sous IBM i, les comptes avec privilèges automatiques sont ceux qui démarrent le gestionnaire de files d'attente et l'ID QMQM.

Vous avez besoin des deux ID, car l'ID utilisateur qui démarre le gestionnaire de files d'attente n'est requis que pour démarrer le système. Une fois en cours d'exécution, les processus du gestionnaire de files d'attente disposent uniquement des droits QMQM.

Windows

Windows

Sous Windows, les comptes avec privilèges complets automatiques sont l'utilisateur du système d'exploitation qui a démarré le gestionnaire de files d'attente, ainsi que l'utilisateur exécutant les processus du gestionnaire de files d'attente principaux, tels que MUSR_MQADMIN si le gestionnaire de files d'attente a été démarré en tant que service Windows .

Lors de l'exécution en mode d'autorisation LDAP, Windows se comporte de manière très similaire aux plateformes UNIX . Il traite des noms abrégés de 12 caractères et des noms distinctifs complets.

Exemple de script

Comme il est utile qu'un groupe puisse effectuer une administration complète sur un gestionnaire de files d'attente, un exemple de script est fourni sur les plateformes UNIX comme suit:

```
MQ_INSTALLATION_PATH/samp/bin/amqauthg.sh
```

Cet exemple utilise deux paramètres:

- Un nom de gestionnaire de files d'attente
- Nom de groupe LDAP

L'exemple traite les commandes setmqaut , en accordant des droits complets sur tous les objets. Il s'agit du même script que celui généré par l'assistant OAM IBM MQ Explorer pour les rôles d'administration. Par exemple, le code démarre:

```
setmqaut -t q -m qmgr -n "**" +alladm +allmqi -g
      groupname
```

Confidentialité des messages

Pour préserver la confidentialité, chiffrez vos messages. Il existe différentes méthodes de chiffrement des messages dans IBM MQ en fonction de vos besoins.

Votre choix de CipherSpec détermine le niveau de confidentialité dont vous disposez.

Si vous avez besoin d'une protection des données de bout en bout au niveau de l'application pour votre infrastructure de messagerie point à point, vous pouvez utiliser Advanced Message Security pour chiffrer les messages ou écrire votre propre exit d'API ou exit de croisement d'API.

Si vous devez chiffrer les messages uniquement lorsqu'ils sont transportés via un canal, car vous disposez d'une sécurité adéquate sur vos gestionnaires de files d'attente, vous pouvez utiliser TLS ou vous pouvez écrire vos propres programmes d'exit de sécurité, d'exit de message ou d'exit d'envoi et de réception.

V 9.1.4 z/OS Si vous devez chiffrer des messages au repos sur un gestionnaire de files d'attente, vous pouvez utiliser le chiffrement de fichier z/OS sur ce gestionnaire de files d'attente.

Pour plus d'informations sur Advanced Message Security, voir «Planification de Advanced Message Security», à la page 106. L'utilisation de TLS avec IBM MQ est décrite à l'adresse «Protocoles de sécurité TLS dans IBM MQ», à la page 24. L'utilisation des programmes d'exit dans le chiffrement des messages est décrite à l'adresse «Implémentation de la confidentialité dans les programmes d'exit utilisateur», à la page 462.

Voir la section Confidentiality for data at rest on IBM MQ for z/OS with data set encryption pour plus d'informations sur le chiffrement des fichiers z/OS .

Tâches associées

Connexion de deux gestionnaires de files d'attente via le protocole TLS

Connexion sécurisée d'un client à un gestionnaire de files d'attente

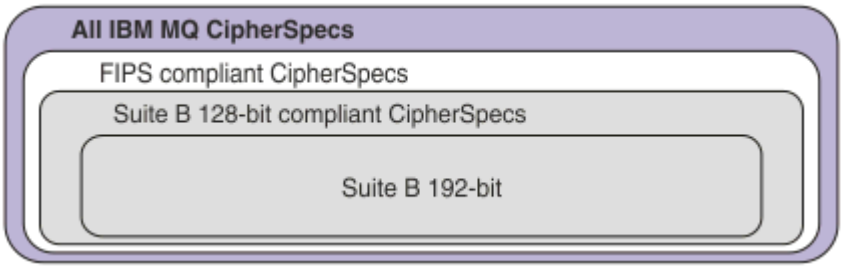
Activation des CipherSpecs

Activez un CipherSpec à l'aide du paramètre **SSLCIPH** dans la commande **DEFINE CHANNEL** MQSC ou dans la commande **ALTER CHANNEL** MQSC.

Certains CipherSpecs que vous pouvez utiliser avec IBM MQ sont conformes à la norme FIPS. Certains des CipherSpecs conformes à la norme FIPS sont également conformes à la norme Suite B, alors que d'autres, tels que TLS_RSA_WITH_AES_256_CBC_SHA, ne le sont pas.

Tous les CipherSpecs conformes à la norme Suite B sont également conformes à la norme FIPS. Tous les CipherSpecs conformes à Suite B appartiennent à deux groupes: 128 bits (par exemple, ECDHE_ECDSA_AES_128_GCM_SHA256) et 192 bits (par exemple, ECDHE_ECDSA_AES_256_GCM_SHA384),

Le diagramme suivant illustre la relation entre ces sous-ensembles:



Depuis la IBM MQ 8.0.0 Fix Pack 3 , le nombre de CipherSpecs pris en charge a été réduit.

V 9.1.1 Pour plus d'informations sur la configuration des CipherSpecspar défaut, voir «Valeurs CipherSpec par défaut activées dans IBM MQ», à la page 438. Vous pouvez également fournir un autre ensemble de CipherSpecs pouvant être utilisés avec les canaux MQ . Voir «Fourniture d'une liste personnalisée de CipherSpecs activés sur Multiplatforms», à la page 439.

Pour plus d'informations sur l'activation des CipherSpecsobsolètes, voir «Activation des CipherSpecs obsolètes sur Multiplatforms», à la page 440 ou «Activation des CipherSpecs obsolètes sous z/OS», à la page 440. Pour la liste des CipherSpecs que vous pouvez réactiver pour les utiliser avec IBM MQ, voir «CipherSpecs obsolètes», à la page 443.

ULW V 9.1.4 Depuis la IBM MQ 9.1.4, IBM MQ prend en charge le protocole de sécurité TLS 1.3 sous UNIX, Linux, and Windows. Pour plus d'informations sur l'utilisation de ces CipherSpecs, voir «Utilisation de TLS 1.3 dans IBM MQ», à la page 438 et «IBM MQ MQI client et TLS 1.3», à la page 438.

CipherSpecs que vous pouvez utiliser avec la prise en charge de TLS dans IBM MQ

Les spécifications de chiffrement que vous pouvez utiliser automatiquement avec le gestionnaire de files d'attente IBM MQ sont répertoriées dans le tableau ci-dessous. Lorsque vous demandez un certificat personnel, vous définissez une taille de clé pour la paire de clé publique et de clé privée. La taille de clé utilisée lors de l'établissement de liaison TLS est la taille stockée dans le certificat, sauf si elle est déterminée par le CipherSpec, comme indiqué dans le tableau.

Tableau 74. CipherSpecs que vous pouvez utiliser avec la prise en charge du protocole TLS dans IBM MQ

Prise en charge des plateformes «1», à la page 437	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 437	Suite B
V 9.1.4 V 9.1.4 **CipherSpecs alias**							
Tous	ANY_TLS13_OR_HIGHER «3», à la page 437 «4», à la page 437 «5», à la page 437	Non disponible	Négocié	Négocié	Négocié	Négocié	Négocié
Tous	ANY_TLS13 «4», à la page 437 «5», à la page 437 «6», à la page 437	Non disponible	TLS 1.3	Négocié	Négocié	Négocié	Négocié
Tous	ANY_TLS12_OR_HIGHER «4», à la page 437 «5», à la page 437 «7», à la page 437	Non disponible	Négocié	Négocié	Négocié	Négocié	Négocié
Tous	ANY_TLS12 «8», à la page 437	Non disponible	TLS 1.2	Négocié	Négocié	Négocié	Négocié
Tous	ANY «9», à la page 437	Non disponible	Négocié	Négocié	Négocié	Négocié	Négocié
V 9.1.4 V 9.1.4 **CipherSpecs pour TLS 1.3**							
Tous	TLS_AES_128_GCM_SHA256 «4», à la page 437	1301	TLS 1.3	GCM	AES-128 avec GCM (128)	Oui	Non
Tous	TLS_AES_256_GCM_SHA384 «4», à la page 437	1302	TLS 1.3	GCM	AES-256 avec GCM (256)	Oui	Non
Tous	TLS_CHACHA20_POLY1305_SHA256 «4», à la page 437	1303	TLS 1.3	POLY1305	CHACHA20 (256)	Non	Non
ULW	TLS_AES_128_CCM_SHA256	1304	TLS 1.3	CBC-MAC	AES-128 avec CTR (128)	Oui	Non
ULW	TLS_AES_128_CCM_8_SHA256 «11», à la page 437	1305	TLS 1.3	CBC-MAC	AES-128 avec CTR (128)	Oui	Non
CipherSpecs pour TLS 1.2							
Tous	TLS_RSA_WITH_AES_128_CBC_SHA256 «10», à la page 437	003C	TLS 1.2	SHA-256	AES (128)	Oui	Non
Tous	TLS_RSA_WITH_AES_256_CBC_SHA256 «10», à la page 437 «12», à la page 437	003D	TLS 1.2	SHA-256	AES (256)	Oui	Non

Tableau 74. CipherSpecs que vous pouvez utiliser avec la prise en charge du protocole TLS dans IBM MQ (suite)

Prise en charge des plateformes «1», à la page 437	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 437	Suite B
Tous	TLS_RSA_WITH_AES_128_GCM_SHA256 «10», à la page 437 «13», à la page 437	009C	TLS 1.2	SHA-256 et AEAD GCM	AES (128)	Oui	Non
Tous	TLS_RSA_WITH_AES_256_GCM_SHA384 «10», à la page 437 «12», à la page 437 «13», à la page 437	009D	TLS 1.2	SHA-384 et AEAD GCM	AES (256)	Oui	Non
Tous	ECDHE_ECDSA_AES_128_CBC_SHA256 «10», à la page 437	C023	TLS 1.2	SHA-256	AES (128)	Oui	Non
Tous	ECDHE_ECDSA_AES_256_CBC_SHA384 «10», à la page 437 «12», à la page 437	C024	TLS 1.2	SHA-384	AES (256)	Oui	Non
Tous	ECDHE_RSA_AES_128_CBC_SHA256 «10», à la page 437	C027	TLS 1.2	SHA-256	AES (128)	Oui	Non
Tous	ECDHE_RSA_AES_256_CBC_SHA384 «10», à la page 437 «12», à la page 437	C028	TLS 1.2	SHA-384	AES (256)	Oui	Non
Multi	ECDHE_ECDSA_AES_128_GCM_SHA256 «12», à la page 437 «13», à la page 437	C02B	TLS 1.2	SHA-256 et AEAD GCM	AES (SHA384)	Oui	128 bits
Multi	ECDHE_ECDSA_AES_256_GCM_SHA384 «12», à la page 437 «13», à la page 437	C02C	TLS 1.2	SHA-384 et AEAD GCM	AES (SHA384)	Oui	192 bits
Tous	ECDHE_RSA_AES_128_GCM_SHA256 «13», à la page 437	C02F	TLS 1.2	SHA-256 et AEAD GCM	AES (128)	Oui	Non
Tous	ECDHE_RSA_AES_256_GCM_SHA384 «12», à la page 437 «13», à la page 437	C030	TLS 1.2	AEAD AES-128 GCM	AES (SHA384)	Oui	Non

Tableau 74. CipherSpecs que vous pouvez utiliser avec la prise en charge du protocole TLS dans IBM MQ (suite)

Prise en charge des plateformes «1», à la page 437	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 437	Suite B

Remarques :

1. Pour la liste des plateformes désignées par chaque icône de plateforme, voir Icônes d'édition et de plateforme dans la documentation du produit.
2. Indique si le CipherSpec est certifié FIPS sur une plateforme certifiée FIPS. Voir la rubrique sur la norme FIPS (Federal Information Processing Standards) pour une explication de la norme FIPS.
3. ULW Le CipherSpec alias ANY_TLS13_OR_HIGHER négocie le niveau supérieur de sécurité que l'extrémité distante autorise mais ne se connecte qu'avec le protocole TLS 1.3 ou une version ultérieure.
4. z/OS Pour utiliser TLS 1.3 ou N'IMPORTE QUEL CipherSpec, sous IBM MQ for z/OS, le système d'exploitation doit être z/OS 2.4 ou une version ultérieure.
5. IBM i Pour que vous puissiez utiliser TLS 1.3 ou le CipherSpec ANY sous IBM i, la version du système d'exploitation sous-jacent doit prendre en charge TLS 1.3. Voir System TLS support for TLSv1.3 pour plus d'informations.
6. ULW Le CipherSpec alias ANY_TLS13 représente un sous-ensemble des CipherSpecs acceptables qui utilisent le protocole TLS 1.3, conformément à la liste dans ce tableau pour chaque plateforme.
7. ULW Le CipherSpec alias ANY_TLS12_OR_HIGHER négocie le niveau supérieur de sécurité que l'extrémité distante autorise mais ne se connecte qu'avec le protocole TLS 1.2 ou une version ultérieure.
8. Le CipherSpec ANY_TLS12 représente un sous-ensemble des CipherSpecs acceptables qui utilisent le protocole TLS 1.2, conformément à la liste dans ce tableau pour chaque plateforme.
9. ULW Le CipherSpec alias ANY négocie le niveau supérieur de sécurité que l'extrémité distante autorise.
10. IBM i Ces CipherSpecs ne sont pas activés sur les systèmes IBM i 7.4 dont la valeur système QSSLCSLCTL a pour valeur *OPSSYS.
11. ULW Ces CipherSpecs utilisent une valeur de contrôle d'intégrité de 8 octets au lieu de 16 octets.
12. Ce CipherSpec ne peut pas être utilisé pour sécuriser une connexion d'IBM MQ Explorer à un gestionnaire de files d'attente, sauf si les fichiers de règles sans restriction appropriés sont appliqués à l'environnement d'exécution Java utilisé par l'explorateur.
13. Windows Linux Suite à une recommandation de GSKit, TLS 1.2 GCM CipherSpecs a une restriction qui signifie qu'après l'envoi d'enregistrements TLS24.5 à l'aide de la même clé de session, la connexion se termine par le message AMQ9288E. Cette restriction GCM est active, quel que soit le mode FIPS utilisé.

    Pour éviter cette erreur, évitez d'utiliser les chiffrements TLS 1.2 GCM , activez la réinitialisation de clé confidentielle ou démarrez votre gestionnaire de files d'attente ou client IBM MQ avec la variable d'environnement GSK_ENFORCE_GCM_RESTRICTION=GSK_FALSE définie. Pour les bibliothèques GSKit , vous devez définir cette variable d'environnement des deux côtés de la connexion et l'appliquer aux connexions client à gestionnaire de files d'attente et de gestionnaire de files d'attente à gestionnaire de files d'attente. Notez que ce paramètre affecte les clients .NET non gérés, mais pas les clients Java ou .NET gérés. Pour plus d'informations, voir AES-GCM cipher restriction.

    Cette restriction ne s'applique pas à IBM MQ for z/OS.

Utilisation de TLS 1.3 dans IBM MQ

ULW V 9.1.4

Depuis la IBM MQ 9.1.4, IBM MQ prend en charge TLS 1.3 sous UNIX, Linux, and Windows. Sur toute installation prise en charge, les nouveaux gestionnaires de files d'attente sont créés avec une entrée dans la strophe SSL du fichier `qm.ini` qui se lit comme suit:

```
SSL:
  AllowTLSV13=TRUE
```

Remarque : Le fichier `qm.ini` se trouve dans le répertoire *<data directory>*`/qmgrs/`*<qmgr name>*.

Si le gestionnaire de files d'attente a été créé à l'aide d'une version de IBM MQ antérieure à IBM MQ 9.1.4, mais qu'il est démarré ultérieurement à l'aide de IBM MQ 9.1.4 ou d'une version ultérieure, la propriété **`AllowTLSV13`** n'est pas définie. Si vous souhaitez activer TLS 1.3, vous devez éditer `qm.ini file` et ajouter la propriété comme indiqué dans l'exemple (y compris la strophe " SSL: s'il n'existe pas encore).

Cette propriété de fichier .ini active TLS 1.3, qui permet l'utilisation de TLS 1.3 CipherSpecs. Conformément à la spécification TLS 1.3, toute tentative de communication avec un CipherSpecfaible, qu'elle soit activée ou non dans IBM MQ , sera rejetée. Les CipherSpecs que TLS 1.3 considère comme faibles sont les CipherSpecs qui répondent à un ou plusieurs des critères suivants:

- Utilise le protocole SSL 3.0 .
- Utilise RC4 ou RC2 comme algorithme de chiffrement.
- A une taille de clé de chiffrement (bit) égale ou inférieure à 112.

Ces restrictions sont signalées par la remarque [10] dans le tableau 1 des CipherSpecsobsolètes.

Si vous devez continuer à utiliser ces CipherSpecs, vous devez désactiver le mode TLS 1.3 . Pour ce faire, éditez le fichier `qm.ini` du gestionnaire de files d'attente et modifiez le paramètre de la propriété **`AllowTLSV13`** en:

```
SSL:
  AllowTLSV13=FALSE
```

Remarque : Avec ce paramètre en place, vous ne pouvez pas utiliser TLS 1.3 CipherSpecs.

IBM MQ MQI client et TLS 1.3

ULW V 9.1.4

Lors de l'utilisation du client IBM MQ MQI client , la valeur de **`AllowTLSV13`** est déduite sauf si elle est explicitement spécifiée dans la strophe SSL du fichier `mqclient.ini` utilisé par l'application.

- Si des CipherSpecs faibles sont activés, **`AllowTLSV13`** est défini sur FALSE et aucun 1.3 CipherSpecs TLS ne peut être utilisé.
- Sinon, **`AllowTLSV13`** est défini sur TRUE et les nouveaux CipherSpecs TLS 1.3 CipherSpecs et alias CipherSpecs peuvent être utilisés.

Valeurs CipherSpec par défaut activées dans IBM MQ

V 9.1.1 Multi

Dans la configuration par défaut, IBM MQ prend en charge le protocole TLS 1.2 et divers algorithmes de cryptographie à l'aide de CipherSpecs. A des fins de compatibilité, IBM MQ peut également être configuré pour utiliser les protocoles SSL 3.0 et TLS 1.0 et un certain nombre d'algorithmes de cryptographie connus pour être faibles ou vulnérables aux vulnérabilités en matière de sécurité. La liste des CipherSpecs qui sont activés dans la configuration par défaut peut changer en appliquant la maintenance.

Il est possible de configurer IBM MQ pour restreindre ou autoriser l'utilisation de CipherSpecs à l'aide des contrôles suivants:

- Autorisez uniquement les CipherSpecs conformes à la norme FIPS 140-2 à l'aide de SSLFIPS.
- ULW Autorisez uniquement les CipherSpecs conformes à NSA Suite B à l'aide de SUITEB.
- ULW Autorisez une liste personnalisée de CipherSpecs à l'aide de **AllowedCipherSpecs** ou de la variable d'environnement **AMQ_ALLOWED_CIPHERS** .
- ULW Autorisez l'utilisation de CipherSpecs obsolètes à l'aide de **AllowWeakCipher** ou de la variable d'environnement **AMQ_SSL_WEAK_CIPHER_ENABLE** .
- z/OS Autorisez l'utilisation de CipherSpecs obsolètes à l'aide d'instructions de définition de données dans le JCL CHINIT.

Remarque : Si vous spécifiez une liste personnalisée de CipherSpecs à l'aide de **AllowedCipherSpecs** ou **AMQ_ALLOWED_CIPHERS** , cela remplace l'activation des CipherSpecsobsolètes. Notez que lorsque vous utilisez des restrictions NSA Suite B ou FIPS 140-2 en combinaison avec une liste CipherSpec personnalisée, vous devez vous assurer que la liste personnalisée contient uniquement des CipherSpecs autorisés par les paramètres Suite B ou FIPS 140-2.

Fourniture d'une liste personnalisée de CipherSpecs activés sur Multiplatforms

V 9.1.1 Multi

Il est possible de fournir un autre ensemble de CipherSpecs pouvant être utilisés avec les canaux IBM MQ , à l'aide de la variable d'environnement **AMQ_ALLOWED_CIPHERS** ou de l'attribut de strophe SSL **AllowedCipherSpecs** du fichier `.ini` . Vous pouvez utiliser ce paramètre pour empêcher les programmes d'écoute IBM MQ d'accepter les demandes de démarrage de canal entrantes, sauf s'ils utilisent l'un des CipherSpecsnommés. Cette fonctionnalité peut être utilisée pour contrôler les CipherSpecs qui sont inclus dans les CipherSpecsANY*.

La variable d'environnement **AMQ_ALLOWED_CIPHERS** ou l'attribut de strophe SSL **AllowedCipherSpecs** accepte:

- Un nom CipherSpec unique, ou
- Liste de noms IBM MQ CipherSpec séparés par des virgules à réactiver, ou
- Valeur spéciale de ALL, représentant tous les CipherSpecs (non recommandé).

Remarque : L'activation de **ALL** CipherSpecs n'est pas recommandée car cela active les protocoles SSL 3.0 et TLS 1.0 et un grand nombre d'algorithmes de cryptographie faibles.

Si ce paramètre est configuré, il remplace la liste CipherSpec par défaut et oblige IBM MQ à ignorer les paramètres de dépréciation de chiffrement faible (voir ci-dessous):

- Les programmes d'écoute IBM MQ n'acceptent que les propositions SSL/TLS qui utilisent l'un des CipherSpecsnommés.
- Les canaux IBM MQ n'autorisent qu'une valeur SSLCIPH vide ou l'un des CipherSpecsnommés.
- La saisie semi-automatique **runmqsc** des valeurs SSLCIPH limite les valeurs de saisie semi-automatique à l'un des noms CipherSpecs.

Par exemple, si vous souhaitez uniquement autoriser les canaux à être définis / modifiés et les programmes d'écoute à accepter ECDHE_RSA_AES_128_GCM_SHA256 ou ECDHE_ECDSA_AES_256_GCM_SHA384 , vous pouvez définir ce qui suit dans le fichier `qm.ini` :

```
SSL:
  AllowedCipherSpecs=ECDHE_RSA_AES_128_GCM_SHA256, ECDHE_ECDSA_AES_256_GCM_SHA384
```

Notez que les chiffrements utilisés par les canaux AMQP ou MQTT peuvent être restreints à l'aide des paramètres de fichier java.security .

Activation des CipherSpecs obsolètes sur Multiplatforms

Multi

Par défaut, vous n'êtes pas autorisé à spécifier un CipherSpec obsolète dans une définition de canal. Si vous tentez de spécifier un CipherSpec obsolète sur Multiplateformes, vous recevez le message AMQ8242: la définition SSLCIPH est incorrecte et PCF renvoie MQRCCF_SSL_CIPHER_SPEC_ERROR.

Vous ne pouvez pas démarrer un canal avec un CipherSpecobsolète. Si vous tentez de le faire avec un CipherSpecobsolète, le système renvoie MQCC_FAILED (2), ainsi qu'un **Reason** de MQRC_SSL_INITIALIZATION_ERROR (2393) au client.

Vous pouvez réactiver une ou plusieurs des CipherSpecs obsolètes pour définir des canaux, lors de l'exécution sur le serveur, en définissant la variable d'environnement **AMQ_SSL_WEAK_CIPHER_ENABLE**.

La variable d'environnement **AMQ_SSL_WEAK_CIPHER_ENABLE** accepte:

- Un nom CipherSpec unique, ou
- Liste de noms IBM MQ CipherSpec séparés par des virgules à réactiver, ou
- Valeur spéciale de ALL, représentant tous les CipherSpecs (non recommandé).

Remarque : La réactivation de ALL CipherSpecs n'est pas recommandée, car cela active les protocoles SSL 3.0 et TLS 1.0 et un grand nombre d'algorithmes de cryptographie faibles.

Par exemple, si vous souhaitez réactiver ECDHE_RSA_RC4_128_SHA256, définissez la variable d'environnement suivante:

```
export AMQ_SSL_WEAK_CIPHER_ENABLE=ECDHE_RSA_RC4_128_SHA256
```

ou modifiez la strophe SSL dans le fichier qm.ini en définissant:

```
SSL:
  AllowTLSV1=Y
  AllowWeakCipherSpec=ECDHE_RSA_RC4_128_SHA256
```

Activation des CipherSpecs obsolètes sous z/OS

z/OS

Par défaut, vous n'êtes pas autorisé à spécifier un CipherSpec obsolète dans une définition de canal. Si vous tentez de spécifier un CipherSpec obsolète sur z/OS, vous recevez le message CSQM102E ou le message CSQX674E.

Pour activer les spécifications de chiffrement faibles (obsolètes), vous devez définir l'instruction de définition de données suivante dans le JCL CHINIT:

```
JCL: //CSQXWEAK DD DUMMY
```

Remarque : Les CipherSpecs obsolètes ne nécessitent pas tous l'utilisation de cette instruction de définition de données. Voir la remarque 11 dans le tableau dans «CipherSpecs obsolètes», à la page 443.

Pour activer le protocole 3.0 SSL obsolète, vous devez également définir l'instruction de définition de données suivante dans le JCL CHINIT:

```
JCL: //CSQXSSL3 DD DUMMY
```

V 9.1.0 Pour activer le protocole 1.0 TLS obsolète, vous devez également définir l'instruction de définition de données suivante dans le JCL CHINIT:

```
JCL: //TLS10ON DD DUMMY
```

Notez que le nom de la carte de définition de données est TLS10ON, ce qui signifie que TLS 1.0 est activé et non TLS100N.

Pour désactiver TLS 1.0 , utilisez l'instruction suivante:

```
JCL: //TLS10OFF DD DUMMY
```

Si vous ne souhaitez pas négocier avec le programme d'écoute à l'aide de spécifications de chiffrement faibles ou rompues, vous devez définir l'instruction de définition de données suivante dans le JCL CHINIT:

```
JCL: //WCIPSOFF DD DUMMY
```

Si vous souhaitez négocier uniquement avec le programme d'écoute à l'aide des spécifications de chiffrement répertoriées dans la liste des spécifications de chiffrement par défaut **System SSL** , vous devez définir l'instruction de définition de données suivante dans le JCL CHINIT:

```
JCL: //GSKDCIPS DD DUMMY
```

Niveau minimal par rapport au niveau fixe CipherSpecs

IBM MQ prend en charge deux types différents de CipherSpecs:

- **Niveau minimal** CipherSpecs sont ceux qui ne définissent pas de limite supérieure, par exemple, ANY, ANY_TLS12_OR_HIGHER ou ANY_TLS13_OR_HIGHER.
- **Niveau fixe** CipherSpecs sont ceux qui identifient un protocole spécifique, par exemple ANY_TLS12 et ANY_TLS13, ou un algorithme spécifique tel que ECDHE_ECDSA_3DES_EDE_CBC_SHA256

Pour optimiser la simplicité de la configuration tout en conservant la sécurité, il est recommandé d'utiliser des CipherSpecs **de niveau minimal** de part et d'autre du canal. Cela permet à vos communications de prendre automatiquement en charge et d'utiliser une version de protocole TLS supérieure lorsque les deux côtés prennent en charge une nouvelle version sans qu'il soit nécessaire de modifier la configuration de l'un ou l'autre côté.

L'utilisation d'un **niveau minimal** CipherSpec du côté du lancement, mais un **niveau fixe** CipherSpec du côté de la réception peut entraîner le rejet de la connexion et l'émission des messages AMQ9631 et AMQ9641 .

Voir «Relation entre les paramètres de l'alias CipherSpec», à la page 447 pour les tableaux contenant des résultats différents pour les paramètres CipherSpec de l'alias.

Concepts associés

«Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45
Cette rubrique fournit des informations sur la façon de choisir les CipherSpecs et les certificats numériques appropriés pour votre règle de sécurité, en soulignant la relation entre les CipherSpecs et les certificats numériques dans IBM MQ.

«CipherSpecs et CipherSuites», à la page 19
Les protocoles de sécurité cryptographique doivent convenir des algorithmes utilisés par une connexion sécurisée. CipherSpecs et CipherSuites définissent des combinaisons spécifiques d'algorithmes.

«Configuration de IBM MQ pour Suite B», à la page 43
IBM MQ peut être configuré pour fonctionner en conformité avec la norme NSA Suite B sur les plateformes Windows, UNIX and Linux .

«FIPS (Federal Information Processing Standards)», à la page 33
Cette rubrique présente le programme FIPS (Federal Information Processing Standards) Cryptomodule Validation Program du US National Institute of Standards and Technology et les fonctions cryptographiques qui peuvent être utilisées sur les canaux TLS.

Tâches associées

Migration des configurations de sécurité existantes en vue de l'utilisation de l'élément CipherSpec ANY_TLS12_OR_HIGHER

Référence associée

De la définition d'un canal

ALTER CHANNEL
Modifier, copier et créer un canal

ULW AES-Restriction de chiffrementGCM

Guide des restrictions imposées sur les chiffrements AES-GCM lorsqu'ils sont utilisés pour la cryptographie TLS. Ces restrictions sont imposées par les organisations IETF et NIST et nécessitent que la même clé de session ne soit pas utilisée pour transférer de manière sécurisée plus de 2 enregistrements TLS$^{24.5}$ lors de l'utilisation des chiffrements AES-GCM .

Pour plus d'informations sur ces restrictions, voir RFC 9325 Section 4.4 Limites d'utilisation des clés et RFC 8446 section 5.5.

IBM MQ n'implémente pas directement la fonctionnalité cryptographique. A la place, plusieurs bibliothèques cryptographiques différentes sont utilisées pour fournir les fonctionnalités TLS et Advanced Message Security . Sur les systèmes d'exploitation Windows, Linuxet AIX , la bibliothèque cryptographique utilisée par IBM MQ est GSKit. Pour les applications, les bibliothèques C et .NET non gérées utilisent GSKit pour la fonctionnalité cryptographique. L'implémentation des algorithmes de chiffrement AES-GCM par GSKit inclut les restrictions spécifiées par le groupe de normes. En outre, ces restrictions sont activées par défaut. Ainsi, la communication TLS IBM MQ , lors de l'utilisation des chiffrements AES-GCM , s'arrête si plus de 2 enregistrements TLS$^{24.5}$ sont transmis à l'aide de la même clé de session.

Remarque : Cette restriction n'est pas présente sur les plateformes IBM i, IBM Z ou IBM MQ for HPE NonStop ou Java/JMS, les applications .NET gérées car des bibliothèques cryptographiques différentes sont utilisées et ces bibliothèques n'ont pas implémenté la même restriction.

Si un canal IBM MQ reste en cours d'exécution pendant une durée suffisante pour que plus de 2 enregistrements TLS$^{24.5}$ soient transmis à l'aide de la même clé de session, la bibliothèque cryptographique sous-jacente met fin à la connexion. Cela provoque l'arrêt du canal et un message d'erreur AMQ9288E est généré. Les applications dont la communication est interrompue de cette manière reçoivent un code retour `MQRC_CONNECTION_BROKEN` de l'opération IBM MQ exécutée.

L'arrêt de la connexion peut être effectué à chaque extrémité de la communication, mais uniquement sur les extrémités qui utilisent GSKit pour la fonctionnalité cryptographique.

Conseils pour l'atténuation de la restriction

Voici quelques options permettant d'empêcher ou de gérer les communications arrêtées en raison de cette restriction:

Utiliser des clients reconnectables

Les applications peuvent être configurées pour tenter automatiquement une reconnexion, en cas d'échec d'une connexion. Cela inclut les connexions qui sont arrêtées en raison de la restriction GCM . Lorsqu'elle est configurée pour la reconnexion, l'application client est restaurée automatiquement à tout point de défaillance et tous les descripteurs permettant d'ouvrir les objets sont restaurés. Cette opération est effectuée sans revenir au code de l'application.

Pour plus d'informations, voir Reconnexion automatique du client.

Définir une valeur de réinitialisation de clé secrète

IBM MQ peut être configuré pour demander une réinitialisation de clé de session après qu'un nombre d'octets configurable a été transféré sur un canal. Une fois cette limite atteinte, IBM MQ demande à la couche cryptographique d'effectuer une réinitialisation de la clé de session, ce qui génère une nouvelle clé de session.

Il est important de noter que la valeur spécifiée est le nombre d'octets transférés, qui est lié à la taille des messages envoyés par IBM MQ. La restriction concerne le nombre d'enregistrements TLS envoyés. Il n'existe pas de mappage direct entre les octets de message et les enregistrements TLS car un enregistrement TLS peut envoyer un nombre maximal d'octets dépendant de l'unité de transmission maximale (MTU) du réseau. Tous les messages envoyés dont la taille est supérieure à cette valeur sont transmis sous la forme de plusieurs enregistrements TLS. La valeur MTU varie d'un

réseau à l'autre. En outre, il existe d'autres raisons pour lesquelles un enregistrement TLS peut avoir besoin d'être envoyé en dehors de la transmission de données de message IBM MQ , par exemple des vérifications de pulsation IBM MQ , des alertes TLS, d'autres messages de protocole IBM MQ . Ces enregistrements TLS supplémentaires sont comptabilisés dans le nombre maximal d'enregistrements TLS, mais ne sont pas comptabilisés dans la valeur de réinitialisation de la clé confidentielle IBM MQ .

La réinitialisation régulière d'une clé de session à l'aide de la réinitialisation de clé secrète peut empêcher l'arrêt du canal en raison de la restriction AES-GCM .

Pour plus d'informations, voir Réinitialisation des clés secrètes SSL et TLS.

V 9.1.4 **Utiliser les spécifications de chiffrement TLS 1.3**

Alors que la restriction AES-GCM est toujours présente lors de l'utilisation du protocole TLS 1.3 , le protocole TLS 1.3 prend en charge l'exécution automatique d'une réinitialisation de clé de session sans qu'il soit nécessaire d'interrompre les communications TLS. Cela permet à GSKit de gérer la réinitialisation de la clé de session lorsqu'elle est nécessaire sans qu' IBM MQ n'ait besoin de demander la réinitialisation d'une clé secrète.

Pour plus d'informations, voir Utilisation de TLS 1.3 dans IBM MQ dans «Activation des CipherSpecs», à la page 434.

Désactivation de la restriction AES-GCM

Si nécessaire, la restriction peut être désactivée en définissant la variable d'environnement **GSK_ENFORCE_GCM_RESTRICTION=GSK_FALSE** pour désactiver la restriction AES-GCM . Cela permet d'envoyer n'importe quel nombre d'enregistrements TLS à l'aide de la même clé de session. Si vous choisissez cette atténuation, la variable d'environnement doit être définie à chaque extrémité de la communication qui utilise GSKit pour les communications sécurisées.

Avertissement : Cette option n'est pas recommandée car, après l'envoi de plus de 2 enregistrements TLS[24.5] , les agresseurs peuvent effectuer une analyse sur les enregistrements envoyés afin de déterminer la clé de session utilisée. Une fois la clé de session déterminée, toutes les communications existantes et futures utilisant cette clé de session sont compromises.

CipherSpecs obsolètes

Liste des CipherSpecs obsolètes que vous pouvez utiliser avec IBM MQ si nécessaire.

Pour plus d'informations sur l'activation des CipherSpecsobsolètes, voir «Activation des CipherSpecs obsolètes sur Multiplatforms», à la page 440 ou «Activation des CipherSpecs obsolètes sous z/OS», à la page 440.

Les CipherSpecs obsolètes que vous pouvez utiliser avec la prise en charge de TLS dans IBM MQ sont répertoriés dans le tableau suivant.

Tableau 75. CipherSpecs dépréciés que vous pouvez réactiver afin de les utiliser avec IBM MQ

Prise en charge des plateformes «1», à la page 446	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 446	Suite B	Mettre à jour si déprécié
CipherSpecs pour SSL 3.0								
IBM i	AES_SHA_US «3», à la page 446	002F	SSL 3.0	SHA-1	AES (128)	Non	Non	9.0.0.0
Tous	DES_SHA_EXPORT «3», à la page 446 «4», à la page 446 «5», à la page 446	0009	SSL 3.0	SHA-1	DES (56)	Non	Non	9.0.0.0

Tableau 75. CipherSpecs dépréciés que vous pouvez réactiver afin de les utiliser avec IBM MQ (suite)

Prise en charge des plateformes «1», à la page 446	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 446	Suite B	Mettre à jour si déprécié
ULW	DES_SHA_EXPORT1024 «3», à la page 446 «6», à la page 446	0062	SSL 3.0	SHA-1	DES (56)	Non	Non	9.0.0.0
ULW	FIPS_WITH_DES_CBC_SHA «3», à la page 446	FEFE	SSL 3.0	SHA-1	DES (56)	Non «7», à la page 446	Non	9.0.0.0
ULW	FIPS_WITH_3DES_EDE_CBC_SHA «3», à la page 446	FEFF	SSL 3.0	SHA-1	3DES (168)	Non «8», à la page 446	Non	9.0.0.1 et 9.0.1
Tous	NULL_MD5 «3», à la page 446	0001	SSL 3.0	MD5	Aucun	Non	Non	9.0.0.1
Tous	NULL_SHA «3», à la page 446	0002	SSL 3.0	SHA-1	Aucun	Non	Non	9.0.0.1
Tous	RC2_MD5_EXPORT «3», à la page 446 «4», à la page 446 «5», à la page 446	0006	SSL 3.0	MD5	RC2 (40)	Non	Non	9.0.0.0
Tous	RC4_MD5_EXPORT «4», à la page 446 «3», à la page 446	0003	SSL 3.0	MD5	RC4 (40)	Non	Non	9.0.0.0
Tous	RC4_MD5_US «3», à la page 446	0004	SSL 3.0	MD5	RC4 (128)	Non	Non	9.0.0.0
Tous	RC4_SHA_US «3», à la page 446 «5», à la page 446	0005	SSL 3.0	SHA-1	RC4 (128)	Non	Non	9.0.0.0
ULW	RC4_56_SHA_EXPORT1024 «3», à la page 446 «6», à la page 446	0064	SSL 3.0	SHA-1	RC4 (56)	Non	Non	9.0.0.0
Tous	TRIPLE_DES_SHA_US «3», à la page 446 «5», à la page 446	000A	SSL 3.0	SHA-1	3DES (168)	Non	Non	9.0.0.1 et 9.0.1
CipherSpecs pour TLS 1.0								
IBM i	TLS_RSA_EXPORT_WITH_RC2_40_MD5 «3», à la page 446	0006	TLS 1.0	MD5	RC2 (40)	Non	Non	9.0.0.0
IBM i	TLS_RSA_EXPORT_WITH_RC4_40_MD5 «3», à la page 446 «4», à la page 446	0003	TLS 1.0	MD5	RC4 (40)	Non	Non	9.0.0.0
Tous	TLS_RSA_WITH_DES_CBC_SHA «3», à la page 446	0009	TLS 1.0	SHA-1	DES (56)	Non «9», à la page 446	Non	9.0.0.0

Tableau 75. CipherSpecs dépréciés que vous pouvez réactiver afin de les utiliser avec IBM MQ (suite)

Prise en charge des plateformes «1», à la page 446	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 446	Suite B	Mettre à jour si déprécié
IBM i	TLS_RSA_WITH_NULL_MD5 «3», à la page 446	0001	TLS 1.0	MD5	Aucun	Non	Non	9.0.0.1
IBM i	TLS_RSA_WITH_NULL_SHA «3», à la page 446	0002	TLS 1.0	SHA-1	Aucun	Non	Non	9.0.0.1
IBM i	TLS_RSA_WITH_RC4_128_MD5 «3», à la page 446	0004	TLS 1.0	MD5	RC4 (128)	Non	Non	9.0.0.0
z/OS ULW	TLS_RSA_WITH_AES_128_CBC_SHA «10», à la page 446	002F	TLS 1.0	SHA-1	AES (128)	Oui	Non	9.0.5
z/OS ULW	TLS_RSA_WITH_AES_256_CBC_SHA «6», à la page 446 «10», à la page 446	0035	TLS 1.0	SHA-1	AES (256)	Oui	Non	9.0.5
Tous	TLS_RSA_WITH_3DES_EDE_CBC_SHA	000A	TLS 1.0	SHA-1	3DES (168)	Oui	Non	9.0.0.1 et 9.0.1
CipherSpecs pour TLS 1.2								
ULW	ECDHE_ECDSA_NULL_SHA256 «3», à la page 446	C006	TLS 1.2	SHA-1	Aucun	Non	Non	9.0.0.1
ULW	ECDHE_ECDSA_RC4_128_SHA256 «3», à la page 446	C007	TLS 1.2	SHA-1	RC4 (128)	Non	Non	9.0.0.0
IBM i ULW	ECDHE_RSA_NULL_SHA256 «3», à la page 446	C010	TLS 1.2	SHA-1	Aucun	Non	Non	9.0.0.1
IBM i ULW	ECDHE_RSA_RC4_128_SHA256 «3», à la page 446	C011	TLS 1.2	SHA-1	RC4 (128)	Non	Non	9.0.0.0
ULW	TLS_RSA_WITH_NULL_NULL «3», à la page 446	0000	TLS 1.2	Aucun	Aucun	Non	Non	9.0.0.1
Tous	TLS_RSA_WITH_NULL_SHA256 «3», à la page 446	003B	TLS 1.2	SHA-256	Aucun	Non	Non	9.0.0.1
ULW	TLS_RSA_WITH_RC4_128_SHA256 «3», à la page 446	0005	TLS 1.2	SHA-1	RC4 (128)	Non	Non	9.0.0.0
ULW	ECDHE_ECDSA_3DES_EDE_CBC_SHA256	C0008	TLS 1.2	SHA-1	3DES (168)	Oui	Non	9.0.0.1 et 9.0.1
IBM i ULW	ECDHE_RSA_3DES_EDE_CBC_SHA256	C012	TLS 1.2	SHA-1	3DES (168)	Oui	Non	9.0.0.1 et 9.0.1

Tableau 75. CipherSpecs dépréciés que vous pouvez réactiver afin de les utiliser avec IBM MQ (suite)

Prise en charge des plateformes «1», à la page 446	Nom du CipherSpec	Code hexadécimal	Protocole utilisé	Intégrité des données	Algorithme de chiffrement (bits de chiffrement)	FIPS «2», à la page 446	Suite B	Mettre à jour si déprécié

Remarques :

1. Pour la liste des plateformes désignées par chaque icône de plateforme, voir Icônes d'édition et de plateforme dans la documentation du produit.
2. Indique si le CipherSpec est certifié FIPS sur une plateforme certifiée FIPS. Voir la rubrique sur la norme FIPS (Federal Information Processing Standards) pour une explication de la norme FIPS.
3. ULW Ces CipherSpecs sont désactivés lorsque TLS 1.3 est activé (via la propriété AllowTLSV13 dans `qm.ini`).

   z/OS Les gestionnaires de files d'attente créés dans IBM MQ for z/OS 9.2.0 ou version ultérieure activent TLS 1.3 par défaut, ce qui désactive ces CipherSpecs. Si nécessaire, vous pouvez activer ces CipherSpecs en désactivant TLS V1.3. Pour ce faire, ajoutez **AllowTLSV13**=*FALSE* à la strophe TransportSecurity du fichier QMINI dans le JCL du gestionnaire de files d'attente. TLS 1.3 n'est pas activé par défaut pour les gestionnaires de files d'attente migrés vers IBM MQ for z/OS 9.2.0 à partir d'une version antérieure. Par conséquent, ces CipherSpecs sont activés.
4. La taille de clé d'établissement de liaison maximale est de 512 bits. Si l'un ou l'autre des certificats échangés lors de l'établissement de liaison SSL a une taille de clé supérieure à 512 bit, une clé temporaire de 512 bits est générée pour l'établissement de liaison.
5. Ces CipherSpec ne sont plus pris en charge par IBM MQ classes for Java et IBM MQ classes for JMS. Pour plus d'informations, voir CipherSpecs et suites de chiffrement SSL/TLS dans IBM MQ classes for Java ou CipherSpecs et suites de chiffrement SSL/TLS dans IBM MQ classes for JMS.
6. La taille de clé d'établissement de liaison maximale est de 1024 bits.
7. Ce CipherSpec a été certifié FIPS 140-2 avant le 19 mai 2007. Le nom `FIPS_WITH_DES_CBC_SHA` est historique et reflète le fait que CipherSpec était précédemment (mais n'est plus) conforme à la norme FIPS. Ce CipherSpec est déprécié et son utilisation est déconseillée.
8. Le nom `FIPS_WITH_3DES_EDE_CBC_SHA` est historique et reflète le fait que CipherSpec était précédemment (mais n'est plus) conforme à la norme FIPS. L'utilisation de ce CipherSpec a été dépréciée.
9. Ce CipherSpec a été certifié FIPS 140-2 avant le 19 mai 2007.
10. z/OS La réactivation de ces CipherSpec ne nécessite pas l'utilisation de l'instruction de définition de données (DD) CSQXWEAK.

Concepts associés

«Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45
Cette rubrique fournit des informations sur la façon de choisir les CipherSpecs et les certificats numériques appropriés pour votre règle de sécurité, en soulignant la relation entre les CipherSpecs et les certificats numériques dans IBM MQ.

Référence associée

De la définition d'un canal
ALTER CHANNEL

Relation entre les paramètres de l'alias CipherSpec

Les tableaux suivants présentent le comportement attendu lorsque TLS1.3 n'est pas activé sur le client, le gestionnaire de files d'attente ou les deux et lorsque TLS1.3 est activé sur le client et le gestionnaire de files d'attente.

Les tableaux suivants montrent la relation entre les différents paramètres d'alias CipherSpec et le résultat attendu. Tableau 76, à la page 447 montre le comportement attendu lorsque TLS 1.3 n'est pas activé sur le client, le serveur ou les deux. Tableau 77, à la page 447 montre le comportement attendu lorsque TLS 1.3 est activé sur le client et le serveur. Dans les deux cas, les CipherSpecs du client sont affichés sur l'axe Y du tableau et les CipherSpecs du serveur sont affichés sur l'axe X du tableau.

Remarque : Lorsque l'entrée indique *Probablement d'échec* , cela est dû au fait que, si le protocole TLS 1.3 ou TLS 1.2 CipherSpec utilisé est le CipherSpec le plus puissant pour le client et le gestionnaire de files d'attente, l'établissement de liaison TLS se résout à l'utiliser et correspond donc à la valeur SSCIPH du canal.

Tableau 76. Comportement attendu lorsque TLS 1.3 n'est pas activé sur le client, le serveur ou les deux

	serveur			
Client	**TLS spécifique 1.2 CipherSpec**	**ANY**	**ANY_ TLS12**	**ANY_TLS12_ OR_HIGHER**
TLS spécifique 1.2 CipherSpec	Connecte	Connecte	Connecte	Connecte
Tout	*Susceptible d'échouer*	Connecte	Connecte	Connecte
ANY_ TLS12	*Susceptible d'échouer*	Connecte	Connecte	Connecte
ANY_TLS12_ OR_HIGHER	*Susceptible d'échouer*	Connecte	Connecte	Connecte

Tableau 77. Comportement attendu lorsque TLS 1.3 est activé sur le client et le serveur

	serveur						
Client	**TLS spécifique 1.2 CipherSpec**	**TLS spécifique 1.3 CipherSpec**	**ANY**	**ANY_TLS 12**	**ANY_TLS 13**	**ANY_TLS12_ OR_SUPERI EUR**	**ANY_TLS13_ OR_SUPERI EUR**
TLS spécifique 1.2 CipherSpec	Connecte	**Echoue**	Connecte	Connecte	**Echoue**	Connecte	**Echoue**
TLS spécifique 1.3 CipherSpec	**Echoue**	Connecte	Connecte	**Echoue**	Connecte	Connecte	Connecte
Tout	**Echoue**	*Susceptible d'échouer*	Connecte	**Echoue**	Connecte	Connecte	Connecte
ANY_TLS12	*Susceptible d'échouer*	**Echoue**	Connecte	Connecte	**Echoue**	Connecte	**Echoue**
ANY_TLS13	**Echoue**	*Susceptible d'échouer*	Connecte	**Echoue**	Connecte	Connecte	Connecte

Tableau 77. Comportement attendu lorsque TLS 1.3 est activé sur le client et le serveur (suite)							
	serveur						
Client	**TLS spécifique 1.2 CipherSpec**	**TLS spécifique 1.3 CipherSpec**	**ANY**	**ANY_TLS 12**	**ANY_TLS 13**	**ANY_TLS12_ OR_SUPERI EUR**	**ANY_TLS13_ OR_SUPERI EUR**
ANY_TLS12_ OR_HIGHER	**Echoue**	*Susceptible d'échouer*	Connecte	**Echoue**	Connecte	Connecte	Connecte
ANY_TLS13_ OR_HIGHER	**Echoue**	*Susceptible d'échouer*	Connecte	**Echoue**	Connecte	Connecte	Connecte

Concepts associés

«Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45
Cette rubrique fournit des informations sur la façon de choisir les CipherSpecs et les certificats numériques appropriés pour votre règle de sécurité, en soulignant la relation entre les CipherSpecs et les certificats numériques dans IBM MQ.

«CipherSpecs et CipherSuites», à la page 19
Les protocoles de sécurité cryptographique doivent convenir des algorithmes utilisés par une connexion sécurisée. CipherSpecs et CipherSuites définissent des combinaisons spécifiques d'algorithmes.

«Activation des CipherSpecs», à la page 434
Activez un CipherSpec à l'aide du paramètre **SSLCIPH** dans la commande **DEFINE CHANNEL** MQSC ou dans la commande **ALTER CHANNEL** MQSC.

Tâches associées

Migration des configurations de sécurité existantes en vue de l'utilisation de l'élément CipherSpec ANY_TLS12_OR_HIGHER

Obtention d'informations sur les CipherSpecs à l'aide de IBM MQ Explorer

Vous pouvez utiliser IBM MQ Explorer pour afficher les descriptions des CipherSpecs.

Utilisez la procédure suivante pour obtenir des informations sur les CipherSpecs dans «Activation des CipherSpecs», à la page 434:

1. Ouvrez IBM MQ Explorer et développez le dossier des **gestionnaires de files d'attente**.
2. Vérifiez que vous avez démarré votre gestionnaire de files d'attente.
3. Sélectionnez le gestionnaire de files d'attente à utiliser et cliquez sur **Canaux**.
4. Cliquez avec le bouton droit de la souris sur le canal que vous souhaitez utiliser et sélectionnez **Propriétés**.
5. Sélectionnez la page de propriétés **SSL** .
6. Dans la liste, sélectionnez le CipherSpec que vous souhaitez utiliser. Une description s'affiche dans la fenêtre située sous la liste.

Alternatives pour la spécification de CipherSpecs

Pour les plateformes sur lesquelles le système d'exploitation fournit la prise en charge TLS, votre système peut prendre en charge de nouveaux CipherSpecs. Vous pouvez spécifier un nouveau CipherSpec avec le paramètre SSLCIPH, mais la valeur que vous fournissez dépend de votre plateforme.

Remarque : Cette section ne s'applique pas aux systèmes UNIX, Linux ou Windows , car les CipherSpecs sont fournis avec le produit IBM MQ , de sorte que les nouveaux CipherSpecs ne deviennent pas disponibles après l'expédition.

Pour les plateformes sur lesquelles le système d'exploitation fournit la prise en charge TLS, votre système peut prendre en charge de nouveaux CipherSpecs qui ne sont pas inclus dans «Activation des CipherSpecs», à la page 434. Vous pouvez spécifier un nouveau CipherSpec avec le paramètre SSLCIPH, mais la valeur que vous fournissez dépend de votre plateforme. Dans tous les cas, la spécification doit correspondre à un CipherSpec TLS valide et pris en charge par la version de TLS exécutée par votre système.

IBM i

Chaîne de deux caractères représentant une valeur hexadécimale.

Pour plus d'informations sur les valeurs autorisées, voir le point 3 de la section Remarques sur l'utilisation de la rubrique Définition des informations sur les caractères pour une session sécurisée.

Avertissement : Vous ne devez pas spécifier de valeur de chiffrement hexadécimal dans SSLCIPH car la valeur ne permet pas d'identifier le chiffrement qui sera utilisé, et le choix du protocole à utiliser est indéterminé. L'utilisation de valeurs de chiffrement hexadécimales peut entraîner des erreurs de non-concordance de CipherSpec .

Vous pouvez utiliser la commande CHGMQMCHL ou CRTMQMCHL pour spécifier la valeur, par exemple:

```
CRTMQMCHL CHLNAME(' channel name ') SSLCIPH(' hexadecimal value ')
```

Vous pouvez également utiliser la commande ALTER QMGR MQSC pour définir le paramètre **SSLCIPH** .

z/OS

Chaîne de quatre caractères représentant une valeur hexadécimale. Les codes hexadécimaux correspondent aux valeurs définies dans le protocole TLS.

Pour plus d'informations, voir Cipher Suite Definitions où figure la liste de toutes les spécifications de chiffrement TLS 1.0, TLS 1.2et TLS 1.3 prises en charge, sous la forme de codes hexadécimaux à 4 chiffres.

Remarques relatives aux clusters IBM MQ

Avec les clusters IBM MQ , il est plus sûr d'utiliser les noms CipherSpec dans «Activation des CipherSpecs», à la page 434. Si vous utilisez une autre spécification, sachez que la spécification peut ne pas être valide sur d'autres plateformes. Pour plus d'informations, voir «SSL/TLS et clusters», à la page 477.

Spécification d'un CipherSpec pour un IBM MQ MQI client

Vous disposez de trois options pour spécifier un CipherSpec pour un IBM MQ MQI client.

Ces options sont les suivantes :

- Utilisation d'une table de définition de canal
- Utilisation de la zone SSLCipherSpec dans la structure MQCD, à l'adresse MQCD_VERSION_7 ou supérieure, sur un appel MQCONNX.
- Utilisation de Active Directory (sur les systèmes Windows avec prise en charge d' Active Directory)

Spécification d'une CipherSuite avec IBM MQ classes for Java et IBM MQ classes for JMS

IBM MQ classes for Java et IBM MQ classes for JMS spécifient les CipherSuites différemment des autres plateformes.

Pour plus d'informations sur la spécification d'une CipherSuite avec IBM MQ classes for Java, voir Transport Layer Security (TLS) support for Java

Pour plus d'informations sur la spécification d'une CipherSuite avec IBM MQ classes for JMS, voir Utilisation de TLS (Transport Layer Security) avec IBM MQ classes for JMS

Spécification d'un CipherSpec pour IBM MQ.NET

Pour IBM MQ.NET , vous pouvez spécifier le CipherSpec à l'aide de la classe MQEnvironment ou à l'aide de la propriété MQC.SSL_CIPHER_SPEC_PROPERTY dans la table de hachage des propriétés de connexion.

Pour plus d'informations sur la spécification d'un CipherSpec pour le client non géré .NET , voir Activation de TLS pour le client .NET non géré

Pour plus d'informations sur la spécification d'un CipherSpec pour le client géré .NET , voir Prise en charge deCipherSpec pour le client .NET géré .

z/OS Utilisation d'AT-TLS avec IBM MQ for z/OS

Application Transparent Transport Layer Security (AT-TLS) fournit une prise en charge TLS pour les applications z/OS sans que ces applications aient à implémenter la prise en charge TLS, ou même à savoir que TLS est utilisé. AT-TLS est disponible uniquement sur z/OS.

AT-TLS peut être utilisé avec toutes les versions de IBM MQ for z/OS.

Avant d'utiliser AT-TLS avec IBM MQ for z/OS, assurez-vous de bien comprendre le «Restrictions», à la page 452 impliqué.

Pour utiliser Application Transparent Transport Layer Security , vous devez définir des instructions de stratégie contenant un ensemble de règles utilisées par z/OS Communications Server pour déterminer les connexions TCP/IP pour lesquelles TLS est activé de manière transparente.

IBM MQ for z/OS possède sa propre implémentation TLS, qui requiert que les canaux aient le paramètre SSLCIPH configuré avec un CipherSpecpris en charge.

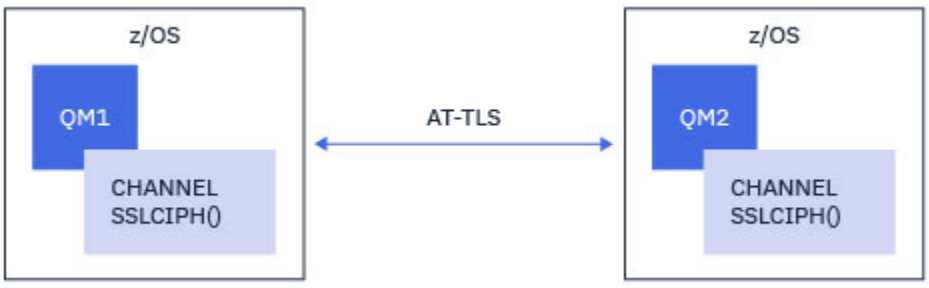
Lorsqu'il décide d'activer TLS sur un canal, l'administrateur IBM MQ peut décider d'utiliser AT-TLS ou IBM MQ TLS. La décision est souvent prise en fonction de l'utilisation d'AT-TLS pour d'autres middlewares ou en raison d'implications en termes de performances. Pour une comparaison de base des performances d'AT-TLS et de IBM MQ TLS, voir MP16: Capacity Planning and Tuning for IBM MQ for z/OS.

Scénarios

L'utilisation d'AT-TLS avec IBM MQ est prise en charge dans les scénarios suivants:

Scénario 1

Entre deux gestionnaires de files d'attente IBM MQ for z/OS où les deux côtés du canal utilisent AT-TLS. Autrement dit, aucun des deux canaux ne spécifie l'attribut SSLCIPH. Cette approche peut être utilisée avec n'importe quel canal de message.

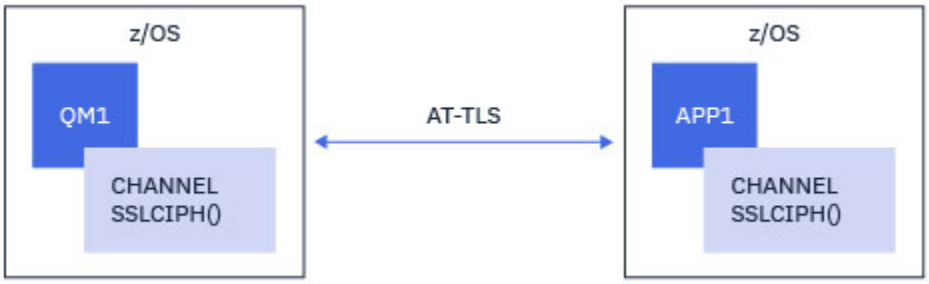


L'implémentation de ce scénario consiste à définir deux règles AT-TLS, une pour chaque côté du canal. Ces règles sont identiques à celles utilisées avec le scénario 3.

Par exemple, si le canal est passé de l'utilisation d'un seul CipherSpec CipherSpec à l'utilisation d'AT-TLS, le canal sortant utilise la règle de «Configuration d'AT-TLS sur un canal sortant vers un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 453 et le canal entrant utilise la règle de «Configuration d'AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 456.

Scénario 2

Entre un gestionnaire de files d'attente IBM MQ for z/OS et une application client IBM MQ Java s'exécutant sur z/OS où les deux côtés du canal utilisent AT-TLS. Autrement dit, ni le canal de connexion serveur, ni le canal de connexion client ne spécifient l'attribut SSLCIPH.

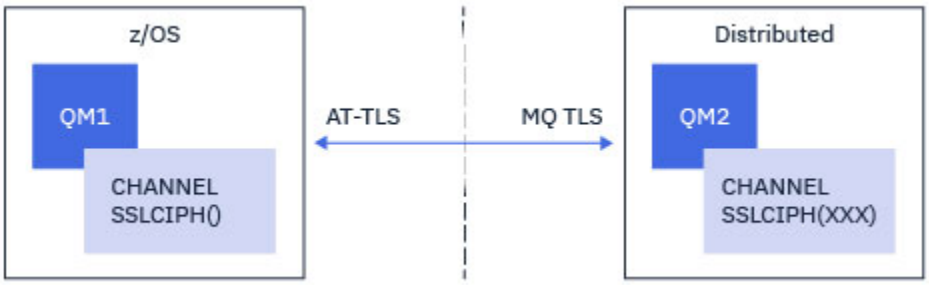


L'implémentation de ce scénario consiste à définir deux règles AT-TLS, une pour chaque côté du canal. Ces règles sont identiques à celles utilisées avec le scénario 3.

Par exemple, si le canal est passé d'un CipherSpec CipherSpec unique à un CipherSpec AT-TLS, le canal de connexion client utilise la règle de «Configuration d'AT-TLS sur un canal sortant vers un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 453 et le canal de connexion serveur utilise la règle de «Configuration d'AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 456.

Scénario 3

Entre un gestionnaire de files d'attente IBM MQ for z/OS et un gestionnaire de files d'attente s'exécutant sous IBM MQ for Multiplatforms, où le gestionnaire de files d'attente IBM MQ for z/OS utilise AT-TLS et le gestionnaire de files d'attente IBM MQ for Multiplatforms utilise IBM MQ TLS. Cela s'applique à tous les types de canaux de transmission de messages autres que les types émetteur et récepteur de cluster.

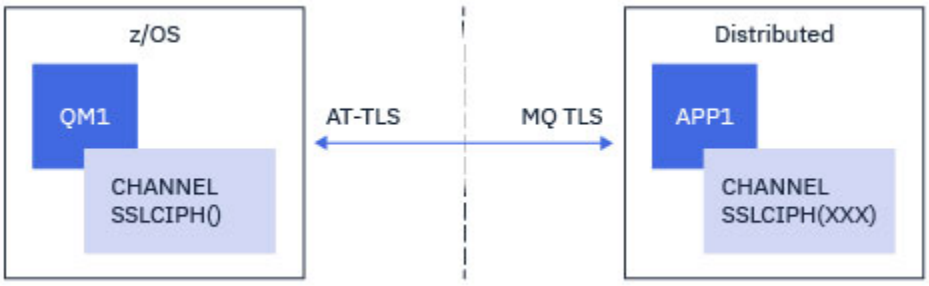


Voir «Configuration d'AT-TLS sur un canal sortant vers un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 453 pour un exemple de configuration AT-TLS pour les canaux sortants du gestionnaire de files d'attente IBM MQ for z/OS vers le gestionnaire de files d'attente IBM MQ for Multiplatforms et «Configuration d'AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 456 pour un exemple de configuration AT-TLS pour les canaux entrants du gestionnaire de files d'attente IBM MQ for Multiplatforms vers le gestionnaire de files d'attente IBM MQ for z/OS .

La même configuration AT-TLS peut être utilisée lorsque les deux gestionnaires de files d'attente sont sous z/OS, mais le gestionnaire de files d'attente à droite n'a pas été configuré pour utiliser AT-TLS.

Scénario 4

Entre un gestionnaire de files d'attente IBM MQ for z/OS et une application client s'exécutant sous IBM MQ for Multiplatforms, où le gestionnaire de files d'attente IBM MQ for z/OS utilise AT-TLS et l'application client utilise IBM MQ TLS en spécifiant l'attribut SSLCIPH avec un seul CipherSpec CipherSpec.



Ce scénario requiert une règle AT-TLS unique qui répond aux mêmes exigences que celles utilisées par un canal de message entrant ; voir «Configuration d'AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec», à la page 456.

La même configuration AT-TLS peut être utilisée lorsque l'application client est une application Java et qu'elle s'exécute également sur z/OS, mais elle n'a pas été configurée pour utiliser AT-TLS.

Restrictions

IBM MQ for z/OS n'est pas sensible à AT-TLS, il existe donc plusieurs restrictions qui s'appliquent aux scénarios précédents:

- AT-TLS combiné à IBM MQ TLS ne fonctionne pas avec les canaux émetteur de cluster et récepteur de cluster.
- Les gestionnaires de files d'attente IBM MQ for z/OS ne savent pas qu'ils utilisent AT-TLS et ne reçoivent pas d'informations de certificat de leur gestionnaire de files d'attente ou client partenaire. Par conséquent, les attributs suivants n'ont aucun effet sur le côté z/OS d'un canal utilisant AT-TLS:
  - Attributs de canal SSLCAUTH et SSLPEER
  - Attribut de gestionnaire de files d'attente SSLRKEYC
  - Attributs SSLPEERMAP des règles CHLAUTH
- L'utilisation de la renégociation de clé secrète TLS requiert que les deux côtés du canal utilisent le protocole TLS IBM MQ . Par conséquent, la renégociation des clés secrètes TLS ne doit pas être activée pour un gestionnaire de files d'attente ou un client IBM MQ for Multiplatforms si vous vous connectez à un gestionnaire de files d'attente IBM MQ for z/OS à l'aide d'AT-TLS.

  Pour désactiver la renégociation de clé secrète TLS pour un gestionnaire de files d'attente, définissez le paramètre SSLRKEYC du gestionnaire de files d'attente sur `0`. Pour un client, définissez le paramètre approprié sur `0` en fonction du type de client. Pour plus de détails sur la procédure à suivre, voir «Réinitialisation des clés secrètes SSL et TLS», à la page 460.

Instructions de configuration AT-TLS

AT-TLS est configuré à l'aide d'un ensemble d'instructions. Les scénarios utilisés dans les scénarios décrits dans cette rubrique sont les suivants:

Règle TTLSRule
: Indique un ensemble de critères pour la mise en correspondance d'une connexion TCP/IP à une configuration TLS. Fait référence aux autres types d'instruction.

TTLSGroupAction
Indique si le `TTLSRule` de référencement est activé ou non.

TTLSEnvironmentAction
Indique la configuration détaillée du `TTLSRule` de référence et fait référence à un certain nombre d'autres instructions.

TTLSKeyringParms
Fait référence au fichier de clés qui doit être utilisé par AT-TLS.

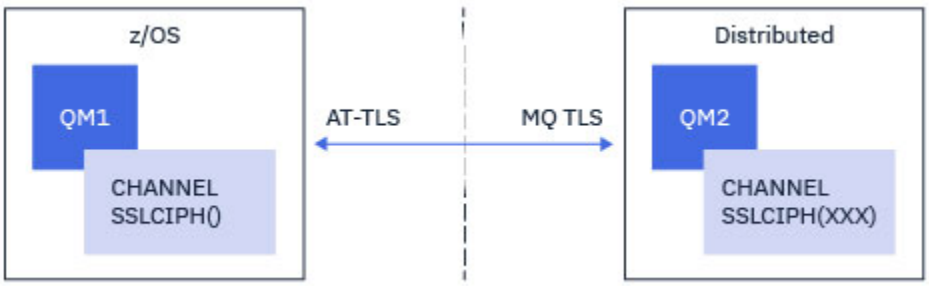
TTLSCipherParms
Définit les suites de chiffrement à utiliser.

TTLSEnvironmentAdvancedParms
Définit les protocoles TLS ou SSL qui sont activés.

Avertissement : Il existe d'autres instructions de règle AT-TLS avec AT-TLS qui ne sont pas documentées ici et qui peuvent être utilisées avec IBM MQ en fonction des besoins. Toutefois, IBM MQ n'a été testé qu'avec les règles décrites dans cette rubrique.

Configuration d'AT-TLS sur un canal sortant vers un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec

Comment configurer AT-TLS sur un canal sortant d'un gestionnaire de files d'attente IBM MQ for z/OS vers un gestionnaire de files d'attente IBM MQ for Multiplatforms . Dans ce cas, le canal du gestionnaire de files d'attente z/OS est un canal émetteur pour lequel l'attribut SSLCIPH n'est pas défini et le canal du gestionnaire de files d'attente nonz/OS est un canal récepteur avec l'attribut SSLCIPH défini sur un seul, nommé CipherSpec.



Dans cet exemple, une paire de canaux émetteur-récepteur existante qui utilise le protocole TLS 1.2 TLS_RSA_WITH_AES_256_GCM_SHA384 CipherSpec va être ajustée pour que le canal émetteur utilise AT-TLS au lieu de IBM MQ TLS.

D'autres protocoles TLS et CipherSpecs peuvent être utilisés en apportant des ajustements mineurs à la configuration. D'autres types de canaux de transmission de messages, à l'exception des canaux émetteur de cluster et récepteur de cluster, pourraient être utilisés sans modification de la configuration AT-TLS.

Procédure

Etape 1: Arrêter le canal

Etape 2: Création et application d'une règle AT-TLS

Vous devez créer les instructions AT-TLS suivantes pour ce scénario:

1. Une instruction TTLSRule pour faire correspondre les connexions sortantes de l'espace adresse de l'initiateur de canal à l'adresse IP et au numéro de port du canal récepteur cible. Ces valeurs doivent correspondre aux informations utilisées dans CONNAME du canal émetteur. Ici, un filtrage supplémentaire a été inclus pour correspondre à un nom de travail d'initiateur de canal spécifique.

```
TTLSRule                        CSQ1-TO-REMOTE
{
  LocalAddr                     ALL
  RemoteAddr                    123.456.78.9
  RemotePortRange               1414
  Jobname                       CSQ1CHIN
  Direction                     OUTBOUND
  TTLSGroupActionRef            CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef      CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}
```

La règle précédente correspond aux connexions à l'adresse IP 123.456.78.9 sur le port 1414 à partir du travail CSQ1CHIN .

Des options de filtrage plus avancées sont décrites dans TTLSRule.

2. Une instruction TTLSGroupAction activant la règle. TTLSRule fait référence à TTLSGroupAction à l'aide de la propriété **TTLSGroupActionRef** .

```
TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}
```

3. Instruction TTLSEnvironmentAction associée à TTLSRule par la propriété **TTLSEnvironmentActionRef** . Un TTLSEnvironmentAction configure l'environnement TLS et spécifie le fichier de clés à utiliser.

```
TTLSEnvironmentAction               CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                     CLIENT
  TTLSKeyringParmsRef               CSQ1-KEYRING
  TTLSCipherParmsRef                CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef   CSQ1-ENVIRONMENT-ADVANCED
}
```

4. Instruction TTLSKeyringParms associée à TTLSEnvironmentAction par la propriété **TTLSKeyringParmsRef** et qui définit le fichier de clés utilisé par AT-TLS.

   Le fichier de clés doit contenir des certificats sécurisés par le gestionnaire de files d'attente nonz/OS distant. Ce fichier de clés peut être défini de la même manière qu'un fichier de clés utilisé par l'initiateur de canal ; voir «Configuration de votre système z/OS pour l'utilisation de TLS», à la page 263.

```
TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}
```

5. Une instruction TTLSCipherParms associée à TTLSEnvironmentAction par la propriété **TTLSCipherParmsRef** .

   Cette instruction doit contenir un nom de suite de chiffrement unique qui doit être l'équivalent du nom IBM MQ CipherSpec utilisé sur le canal récepteur cible.

   Remarque : Les noms de suite de chiffrement AT-TLS ne correspondent pas nécessairement aux noms IBM MQ CipherSpec . Toutefois, il est possible de trouver le nom de la suite de chiffrement AT-TLS qui correspond à un nom IBM MQ CipherSpec en recherchant le nom IBM MQ CipherSpec dans le tableau suivant et en croisant la colonne de code à quatre caractères avec la colonne de caractères développée du tableau 2 dans la rubrique TTLSCipherParms .

Tableau 78. Conversion des codes à quatre caractères en noms CipherSpec

Code à quatre caractères	Protocole	Activé par défaut	Nom du CipherSpec
0001	SSL 3.0	Non	NULL_MD5
0002	SSL 3.0	Non	NULL_SHA
0003	SSL 3.0	Non	RC4_MD5_EXPORT
0004	SSL 3.0	Non	RC4_MD5_US
0005	SSL 3.0	Non	RC4_SHA_US
0006	SSL 3.0	Non	RC2_MD5_EXPORT
0008	SSL 3.0	Non	DES_SHA_EXPORT
0009	TLS 1.0	Oui	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	Non	TRIPLE_DES_SHA_US
000A	TLS 1.0	Oui	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Oui	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Oui	TLS_RSA_WITH_AES_256_CBC_SHA
003B	TLS 1.2	Oui	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Oui	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Oui	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Oui	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Oui	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Oui	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Oui	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Une instruction TTLSEnvironmentAdvancedParms est associée à TTLSEnvironmentAction par la propriété **TTLSEnvironmentAdvancedParmsRef** .

   Cette instruction peut être utilisée pour spécifier les protocoles SSL et TLS qui sont activés. Avec IBM MQ , vous devez activer uniquement le protocole unique qui correspond au nom de l'algorithme de cryptographie utilisé dans l'instruction TTLSCipherParms .

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3           OFF
  TLSv1           OFF
  TLSv1.1         OFF
  SecondaryMap    OFF
  TLSv1.2         ON
  TLSv1.3         OFF
}
```

L'ensemble complet des instructions est le suivant et doit être appliqué à l'agent de règles:

```
TTLSRule                      CSQ1-TO-REMOTE
{
  LocalAddr                   ALL
  RemoteAddr                  123.456.78.9
  RemotePortRange             1414
  Jobname                     CSQ1CHIN
  Direction                   OUTBOUND
  TTLSGroupActionRef          CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef    CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}

TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}

TTLSEnvironmentAction             CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                   CLIENT
  TTLSKeyringParmsRef             CSQ1-KEYRING
  TTLSCipherParmsRef              CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}

TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3          OFF
  TLSv1          OFF
  TLSv1.1        OFF
  SecondaryMap   OFF
  TLSv1.2        ON
  TLSv1.3        OFF
}
```

Etape 3: Suppression de SSLCIPH du canal z/OS

Supprimez le CipherSpec du canal z/OS à l'aide de la commande suivante:

```
ALTER CHANNEL(channel-name) CHLTYPE(SDR) SSLCIPH(' ')
```

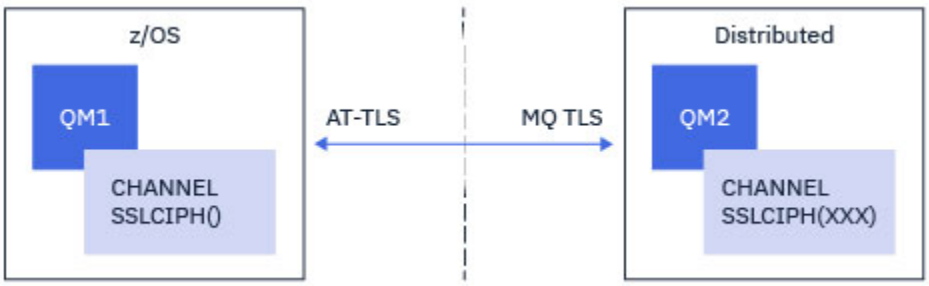
Etape 4: Démarrez le canal

Une fois le canal démarré, il utilise une combinaison d'AT-TLS et IBM MQ TLS.

Avertissement : Les instructions AT-TLS précédentes ne sont qu'une configuration minimale. Il existe d'autres instructions de règle AT-TLS avec AT-TLS qui ne sont pas documentées ici et qui peuvent être utilisées avec IBM MQ en fonction des besoins. Toutefois, IBM MQ n'a été testé qu'avec les règles décrites.

Configuration d'AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms à l'aide d'un seul CipherSpec CipherSpec

Comment configurer AT-TLS sur un canal entrant à partir d'un gestionnaire de files d'attente IBM MQ for Multiplatforms vers un gestionnaire de files d'attente IBM MQ for z/OS . Dans ce cas, le canal du gestionnaire de files d'attente z/OS est un canal récepteur dont l'attribut SSLCIPH n'est pas défini et le canal du gestionnaire de files d'attente nonz/OS est un canal émetteur dont l'attribut SSLCIPH est défini sur un seul, nommé CipherSpec.



Dans cet exemple, une paire de canaux émetteur-récepteur existante, qui utilise TLS 1.2 TLS_RSA_WITH_AES_256_GCM_SHA384 CipherSpec , va être ajustée pour que le canal récepteur utilise AT-TLS au lieu de IBM MQ TLS.

D'autres protocoles TLS et CipherSpecs peuvent être utilisés en apportant des ajustements mineurs à la configuration. D'autres types de canaux de transmission de messages, à l'exception des canaux émetteur de cluster et récepteur de cluster, pourraient être utilisés sans modification de la configuration AT-TLS.

Procédure

Etape 1: Arrêter le canal

Etape 2: Création et application d'une règle AT-TLS

Vous devez créer les instructions AT-TLS suivantes pour ce scénario:

1. Une instruction TTLSRule permettant de faire correspondre les connexions entrantes à l'espace adresse de l'initiateur de canal à partir de l'adresse IP du canal émetteur. Ici, un filtrage supplémentaire a été inclus pour correspondre à un nom de travail d'initiateur de canal spécifique.

```
TTLSRule                        REMOTE-TO-CSQ1
{
  LocalAddr                     ALL
  LocalPortRange                1414
  RemoteAddr                    123.456.78.9
  Jobname                       CSQ1CHIN
  Direction                     INBOUND
  TTLSGroupActionRef            CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef      CSQ1-INBOUND-ENVIRONMENT-ACTION
}
```

   La règle précédente correspond aux connexions entrant dans le travail CSQ1CHIN sur le port local 1414 à partir de l'adresse IP distante 123.456.78.9.

   Des options de filtrage plus avancées sont décrites dans TTLSRule.

2. Une instruction TTLSGroupAction activant la règle. TTLSRule fait référence à TTLSGroupAction à l'aide de la propriété **TTLSGroupActionRef** .

```
TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}
```

3. Une instruction TTLSEnvironmentAction est associée à TTLSRule par la propriété **TTLSEnvironmentActionRef** . Un TTLSEnvironmentAction configure l'environnement TLS et spécifie le fichier de clés à utiliser.

```
TTLSEnvironmentAction                 CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                       SERVER
  TTLSKeyringParmsRef                 CSQ1-KEYRING
  TTLSCipherParmsRef                  CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef     CSQ1-ENVIRONMENT-ADVANCED
}
```

AT-TLS offre la possibilité de fournir une authentification mutuelle, ce qui est l'équivalent de l'utilisation de l'attribut de canal SSLCAUTH. Pour cela, une instruction TTLSEnvironmentAction est associée à la valeur **HandshakeRole** *ServerWithClientAuth* pour l'instruction TTLSEnvironmentAction entrante.

4. Une instruction TTLSKeyringParms est associée à TTLSEnvironmentAction par la propriété **TTLSKeyringParmsRef** et définit le fichier de clés utilisé par AT-TLS.

   Le fichier de clés doit contenir des certificats sécurisés par le gestionnaire de files d'attente nonz/OS distant. Ce fichier de clés peut être défini de la même manière qu'un fichier de clés utilisé par l'initiateur de canal ; voir «Configuration de votre système z/OS pour l'utilisation de TLS», à la page 263.

```
TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}
```

5. Une instruction TTLSCipherParms associée à TTLSEnvironmentAction par la propriété **TTLSCipherParmsRef** .

   Cette instruction doit contenir un nom de suite de chiffrement unique qui doit être l'équivalent du nom IBM MQ CipherSpec utilisé sur le canal émetteur distant.

   Remarque : Les noms de suite de chiffrement AT-TLS ne correspondent pas nécessairement aux noms IBM MQ CipherSpec . Toutefois, il est possible de trouver le nom de la suite de chiffrement AT-TLS qui correspond à un nom IBM MQ CipherSpec en recherchant le nom IBM MQ CipherSpec dans le tableau suivant et en croisant la colonne de code à quatre caractères avec la colonne de caractères développée du tableau 2 dans la rubrique TTLSCipherParms .

Tableau 79. Conversion des codes à quatre caractères en noms CipherSpec

Code à quatre caractères	Protocole	Activé par défaut	Nom du CipherSpec
0001	SSL 3.0	Non	NULL_MD5
0002	SSL 3.0	Non	NULL_SHA
0003	SSL 3.0	Non	RC4_MD5_EXPORT
0004	SSL 3.0	Non	RC4_MD5_US
0005	SSL 3.0	Non	RC4_SHA_US
0006	SSL 3.0	Non	RC2_MD5_EXPORT
0008	SSL 3.0	Non	DES_SHA_EXPORT
0009	TLS 1.0	Oui	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	Non	TRIPLE_DES_SHA_US
000A	TLS 1.0	Oui	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Oui	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Oui	TLS_RSA_WITH_AES_256_CBC_SHA

Tableau 79. Conversion des codes à quatre caractères en noms CipherSpec (suite)

Code à quatre caractères	Protocole	Activé par défaut	Nom du CipherSpec
003B	TLS 1.2	Oui	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Oui	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Oui	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Oui	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Oui	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Oui	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Oui	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Une instruction TTLSEnvironmentAdvancedParms est associée à TTLSEnvironmentAction par la propriété **TTLSEnvironmentAdvancedParmsRef** .

   Cette instruction peut être utilisée pour spécifier les protocoles SSL et TLS qui sont activés. Avec IBM MQ , vous devez activer uniquement le protocole unique qui correspond au nom de l'algorithme de cryptographie utilisé dans l'instruction TTLSCipherParms .

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3           OFF
  TLSv1           OFF
  TLSv1.1         OFF
  SecondaryMap    OFF
  TLSv1.2         ON
  TLSv1.3         OFF
}
```

L'ensemble complet des instructions est le suivant et doit être appliqué à l'agent de règles:

```
TTLSRule                        REMOTE-TO-CSQ1
{
  LocalAddr                     ALL
  LocalPortRange                1414
  RemoteAddr                    123.456.78.9
  Jobname                       CSQ1CHIN
  Direction                     INBOUND
  TTLSGroupActionRef            CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef      CSQ1-INBOUND-ENVIRONMENT-ACTION
}

TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}

TTLSEnvironmentAction             CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                   CLIENT
  TTLSKeyringParmsRef             CSQ1-KEYRING
  TTLSCipherParmsRef              CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}

TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3          OFF
  TLSv1          OFF
  TLSv1.1        OFF
  SecondaryMap   OFF
  TLSv1.2        OFF
  TLSv1.3        ON
}
```

Etape 3: Suppression de SSLCIPH du canal z/OS

Supprimez le CipherSpec du canal z/OS à l'aide de la commande suivante:

```
ALTER CHANNEL(channel-name) CHLTYPE(RCVR) SSLCIPH(' ')
```

Etape 4: Démarrez le canal

Une fois le canal démarré, il utilise une combinaison d'AT-TLS et IBM MQ TLS.

Avertissement : Les instructions AT-TLS précédentes ne sont qu'une configuration minimale. Il existe d'autres instructions de règle AT-TLS avec AT-TLS qui ne sont pas documentées ici et qui peuvent être utilisées avec IBM MQ en fonction des besoins. Toutefois, IBM MQ n'a été testé qu'avec les règles décrites.

Réinitialisation des clés secrètes SSL et TLS

IBM MQ prend en charge la réinitialisation des clés secrètes sur les gestionnaires de files d'attente et les clients.

Les clés secrètes sont réinitialisées lorsqu'un nombre spécifié d'octets chiffrés de données ont transité sur le canal. Si les pulsations de canal sont activées, la clé secrète est réinitialisée avant que les données ne soient envoyées ou reçues à la suite d'une pulsation de canal.

La valeur de réinitialisation de clé est toujours définie par le côté initiateur du canal IBM MQ .

Gestionnaire de files d'attente

Pour un gestionnaire de files d'attente, utilisez la commande **ALTER QMGR** avec le paramètre **SSLRKEYC** pour définir les valeurs utilisées lors de la renégociation de clé.

Sous IBM i, utilisez **CHGMQM** avec le paramètre **SSLRSTCNT** .

MQI Client

Par défaut, les clients MQI ne renégocient pas la clé secrète. Vous pouvez faire en sorte qu'un client MQI renégocie la clé de trois manières. Dans la liste suivante, les méthodes sont affichées par ordre de priorité. Si vous spécifiez plusieurs valeurs, la valeur de priorité la plus élevée est utilisée.

1. En utilisant la zone KeyResetCount dans la structure MQSCO sur un appel MQCONNX
2. A l'aide de la variable d'environnement MQSSLRESET
3. En définissant l'attribut SSLKeyResetCount dans le fichier de configuration du client MQI

Ces variables peuvent être définies sur un entier compris entre 0 et 999 999 999, représentant le nombre d'octets non chiffrés envoyés et reçus dans une conversation TLS avant que la clé secrète TLS ne soit renégociée. La valeur 0 indique que les clés secrètes TLS ne sont jamais renégociées. Si vous spécifiez un nombre de réinitialisations de clé confidentielle TLS compris entre 1 octet et 32 Ko, les canaux TLS utiliseront un nombre de réinitialisations de clé confidentielle de 32 Ko. Cela permet d'éviter un nombre excessif de réinitialisations de clé qui se produiraient pour les petites valeurs de réinitialisation de clé confidentielle TLS.

Si une valeur supérieure à zéro est spécifiée et que les pulsations de canal sont activées pour le canal, la clé secrète est également renégociée avant que les données de message ne soient envoyées ou reçues à la suite d'une pulsation de canal.

Nombre d'octets jusqu'à ce que la prochaine renégociation de clé secrète soit réinitialisée après chaque renégociation réussie.

Pour plus de détails sur la structure MQSCO, voir KeyResetCount (MQLONG). Pour plus de détails sur MQSSLRESET, voir MQSSLRESET. Pour plus d'informations sur l'utilisation de TLS dans le fichier de configuration du client, voir Strophe SSL du fichier de configuration du client.

Java

Pour IBM MQ classes for Java, une application peut réinitialiser la clé secrète de l'une des manières suivantes:

- En définissant la zone sslResetCount dans la classe MQEnvironment.
- En définissant la propriété d'environnement MQC.SSL_RESET_COUNT_PROPERTY dans un objet Hashtable. L'application affecte ensuite la table de hachage à la zone `properties` de la classe MQEnvironment ou transmet la table de hachage à un objet MQQueueManager sur son constructeur.

Si l'application utilise plusieurs de ces méthodes, les règles de priorité habituelles s'appliquent. Voir Class com.ibm.mq.MQEnvironment pour les règles de priorité.

La valeur de la zone sslResetCount ou de la propriété d'environnement MQC.SSL_RESET_COUNT_PROPERTY représente le nombre total d'octets envoyés et reçus par le code client IBM MQ classes for Java avant la renégociation de la clé secrète. Le nombre d'octets envoyés est le nombre avant chiffrement et le nombre d'octets reçus est le nombre après déchiffrement. Le nombre d'octets inclut également les informations de contrôle envoyées et reçues par le client IBM MQ classes for Java .

Si le nombre de réinitialisations est égal à zéro, ce qui correspond à la valeur par défaut, la clé secrète n'est jamais renégociée. Le nombre de réinitialisations est ignoré si CipherSuite n'est pas spécifié.

JMS

Pour IBM MQ classes for JMS, la propriété SSLRESETCOUNT représente le nombre total d'octets envoyés et reçus par une connexion avant que la clé secrète utilisée pour le chiffrement ne soit renégociée. Le nombre d'octets envoyés est le nombre avant chiffrement et le nombre d'octets reçus est le nombre après déchiffrement. Le nombre d'octets inclut également les informations de contrôle envoyées et reçues par IBM MQ classes for JMS. Par exemple, pour configurer un objet ConnectionFactory pouvant être utilisé pour créer une connexion via un canal MQI activé par TLS avec une clé secrète renégociée après la transmission de 4 Mo de données, exécutez la commande suivante à JMSAdmin:

```
ALTER CF(my.cf) SSLRESETCOUNT(4194304)
```

Si la valeur de SSLRESETCOUNT est zéro, qui est la valeur par défaut, la clé secrète n'est jamais renégociée. La propriété SSLRESETCOUNT est ignorée si SSLCIPHERSUITE n'est pas défini.

.NET

Pour les clients non gérés .NET , la propriété d'entier SSLKeyResetindique le nombre d'octets non chiffrés envoyés et reçus dans une conversation TLS avant que la clé secrète ne soit renégociée.

Pour plus d'informations sur l'utilisation des propriétés d'objet dans IBM MQ classes for .NET, voir Obtention et définition des valeurs d'attribut.

Pour les clients gérés .NET , la classe SSLStream ne prend pas en charge la réinitialisation / renégociation des clés secrètes. Toutefois, pour être cohérent avec les autres clients IBM MQ , le client IBM MQ géré .NET permet aux applications de définir le nombre SSLKeyReset. Pour plus d'informations, voir Secret key reset or renégociation.

XMS .NET

Pour les clients XMS .NET non gérés, voir Connexions sécurisées à un gestionnaire de files d'attente IBM MQ.

Référence associée

ALTER QMGR

DISPLAY QMGR

Modification du gestionnaire de files d'attente de messages (CHGMQM)

Afficher le gestionnaire de files d'attente de messages (DSPMQM)

Implémentation de la confidentialité dans les programmes d'exit utilisateur

Implémentation de la confidentialité dans les exits de sécurité

Les exits de sécurité peuvent jouer un rôle dans le service de confidentialité en générant et en distribuant la clé symétrique pour le chiffrement et le déchiffrement des données qui circulent sur le canal. Une technique courante pour ce faire utilise la technologie PKI.

Un exit de sécurité génère une valeur de données aléatoire, le chiffre à l'aide de la clé publique du gestionnaire de files d'attente ou de l'utilisateur que l'exit de sécurité partenaire représente et envoie les données chiffrées à son partenaire dans un message de sécurité. L'exit de sécurité partenaire déchiffre la valeur de données aléatoires avec la clé privée du gestionnaire de files d'attente ou de l'utilisateur qu'il représente. Chaque exit de sécurité peut désormais utiliser la valeur de données aléatoires pour dériver la clé symétrique indépendamment l'une de l'autre en utilisant un algorithme connu des deux. Ils peuvent également utiliser la valeur de données aléatoire comme clé.

Si le premier exit de sécurité n'a pas authentifié son partenaire à ce moment-là, le message de sécurité suivant envoyé par le partenaire peut contenir une valeur attendue chiffrée avec la clé symétrique. Le premier exit de sécurité peut désormais authentifier son partenaire en vérifiant que l'exit de sécurité du partenaire a pu chiffrer correctement la valeur attendue.

Les exits de sécurité peuvent également utiliser cette opportunité pour convenir de l'algorithme de chiffrement et de déchiffrement des données qui circulent sur le canal, si plusieurs algorithmes sont disponibles.

Implémentation de la confidentialité dans les exits de message

Un exit de message à l'extrémité émettrice d'un canal peut chiffrer les données d'application dans un message et un autre exit de message à l'extrémité réceptrice du canal peut déchiffrer les données. Pour des raisons de performances, un algorithme de clé symétrique est normalement utilisé à cette fin. Pour plus d'informations sur la façon dont la clé symétrique peut être générée et distribuée, voir «Implémentation de la confidentialité dans les programmes d'exit utilisateur», à la page 462.

Les en-têtes d'un message, tels que l'en-tête de file d'attente de transmission, MQXQH, qui inclut le descripteur de message imbriqué, ne doivent pas être chiffrés par un exit de message. En effet, la conversion des données des en-têtes de message a lieu soit après l'appel d'un exit de message à l'extrémité émettrice, soit avant l'appel d'un exit de message à l'extrémité réceptrice. Si les en-têtes sont chiffrés, la conversion des données échoue et le canal s'arrête.

Implémentation de la confidentialité dans les exits d'envoi et de réception

Les exits d'envoi et de réception peuvent être utilisés pour chiffrer et déchiffrer les données qui circulent sur un canal. Ils sont plus appropriés que les exits de message pour fournir ce service pour les raisons suivantes:

- Sur un canal de message, les en-têtes de message peuvent être chiffrés ainsi que les données d'application dans les messages.
- Les exits d'envoi et de réception peuvent être utilisés sur les canaux MQI ainsi que sur les canaux de message. Les paramètres des appels MQI peuvent contenir des données d'application sensibles qui doivent être protégées alors qu'elles circulent sur un canal MQI. Vous pouvez donc utiliser les mêmes exits d'émission et de réception sur les deux types de canaux.

Implémentation de la confidentialité dans l'exit d'API et l'exit de croisement d'API

Les données d'application d'un message peuvent être chiffrées par une API ou un exit de croisement d'API lorsque le message est inséré par l'application émettrice et déchiffré par un deuxième exit lorsque le message est extrait par l'application réceptrice. Pour des raisons de performances, un algorithme de clé symétrique est généralement utilisé à cette fin. Toutefois, au niveau de l'application, où de nombreux utilisateurs peuvent s'envoyer des messages les uns aux autres, le problème est de s'assurer que seul le destinataire prévu d'un message est en mesure de déchiffrer le message. Une solution consiste à utiliser une clé symétrique différente pour chaque paire d'utilisateurs qui s'envoient des messages. Mais cette solution peut être difficile et longue à administrer, en particulier si les utilisateurs appartiennent à des organisations différentes. Un moyen standard de résoudre ce problème est appelé *enveloppage numérique* et utilise la technologie PKI.

Lorsqu'une application place un message dans une file d'attente, une API ou un exit de croisement d'API génère une clé symétrique aléatoire et utilise la clé pour chiffrer les données d'application dans le message. L'exit chiffre la clé symétrique avec la clé publique du destinataire prévu. Il remplace ensuite les données d'application du message par les données d'application chiffrées et la clé symétrique chiffrée. De cette manière, seul le récepteur visé peut déchiffrer la clé symétrique et donc les données d'application. Si un message chiffré a plus d'un récepteur prévu possible, l'exit peut chiffrer une copie de la clé symétrique pour chaque récepteur prévu.

Si différents algorithmes de chiffrement et de déchiffrement des données d'application sont disponibles, l'exit peut inclure le nom de l'algorithme qu'il a utilisé.

V 9.1.4 z/OS Confidentialité des données au repos sur IBM MQ for z/OS avec chiffrement de fichier

IBM MQ for z/OS peut renforcer les données client et de configuration en écrivant les données dans les fichiers journaux actifs, les fichiers journaux d'archivage, les ensembles de pages, les fichiers d'amorçage (BSDS) et V 9.1.5 les fichiers de messages partagés (SMDS).

z/OS fournit un chiffrement efficace et basé sur des règles des fichiers. IBM MQ for z/OS prend en charge le chiffrement des fichiers z/OS pour:

- Fichiers journaux actifs ; voir la remarque «1», à la page 464
- Fichiers journaux archivés ; voir la remarque «2», à la page 464
- Ensembles de pages ; voir la remarque «1», à la page 464
- Fichier d'amorce ; voir la remarque «2», à la page 464
- Ensembles de données CSQINP* ; voir la remarque «2», à la page 464
- V 9.1.5 Fichier SMDS ; voir la remarque «3», à la page 464

Cela assure la confidentialité des données au repos sur un gestionnaire de files d'attente z/OS individuel.

Remarques :

1. Depuis la IBM MQ 9.1.4, IBM MQ for z/OS prend en charge le chiffrement des fichiers z/OS pour les journaux actifs et les ensembles de pages.
2. Le chiffrement de fichier pour les journaux d'archivage, les fichiers BSDS et les fichiers CSQINP* est pris en charge sur toutes les versions de IBM MQ for z/OS.
3. V 9.1.5 Depuis la IBM MQ 9.1.5, IBM MQ for z/OS prend en charge le chiffrement des fichiers z/OS pour le fichier SMDS.
4. IBM MQ Advanced Message Security fournit un autre mécanisme de protection des données au repos. En outre, AMS protège également les données en mémoire et en vol

Pour plus d'informations sur le chiffrement des fichiers z/OS , voir Utilisation des améliorations du chiffrement des fichiers z/OS .

La configuration du chiffrement des fichiers z/OS est hors du contrôle de IBM MQ for z/OS. Les paramètres de chiffrement prennent effet lorsque le fichier est créé.

Cela signifie que tous les fichiers existants doivent être recréés avant qu'une nouvelle règle de chiffrement de fichier puisse être utilisée.

IBM MQ for z/OS peut s'exécuter avec un mélange de fichiers chiffrés et non chiffrés, mais une configuration standard chiffrera tous les fichiers utilisés, ou aucun.

V 9.1.4 z/OS Présentation des étapes de chiffrement d'un fichier IBM MQ for z/OS

Comment chiffrer un fichier IBM MQ for z/OS .

Avant de commencer

Vous devez vous assurer que vous avez configuré correctement le chiffrement des fichiers z/OS dans votre entreprise. Si vous configurez le chiffrement des ensembles de données dans un groupe de partage de files d'attente, vous devez configurer le chiffrement des ensembles de données z/OS pour le partage de données.

Remarque : Un fichier chiffré z/OS doit être un fichier au format étendu.

Procédure

1. Configurez la clé de chiffrement et `key-label` dans RACF à utiliser pour chiffrer le jeu de données.
2. Créez un profil pour `key-label` dans la classe RACF CSFKEYS.
3. Accordez un accès en lecture (READ) à l'ID utilisateur du gestionnaire de files d'attente et à tous les autres ID utilisateur qui ont besoin d'accéder aux données chiffrées.

   Il peut s'agir d'ID utilisateur utilisés pour exécuter des utilitaires d'impression sur le fichier. Par exemple, l'utilisateur exécutant CSQUTIL SCOPY doit déchiffrer l'ensemble de pages approprié.
4. Associez le chiffrement `key-label` au nom de fichier.

   Pour ce faire, vous pouvez utiliser une classe de données SMS ou un segment DFP RACF pour le nom de fichier ou le qualificatif de haut niveau.

   Vous pouvez également associer le `key-label` au fichier lorsque ce dernier est alloué.
5. Renommez tout fichier existant à l'aide de l'instruction IDCAMS ALTER.
6. Réallouez le fichier avec les attributs appropriés.
7. Copiez le contenu du fichier renommé dans le nouveau fichier à l'aide d'IDCAMS REPRO.

   Les données sont chiffrées par l'action de copie dans le jeu de données.
8. Répétez les étapes «4», à la page 465 à «6», à la page 465 pour tous les autres fichiers qui doivent être chiffrés.

V 9.1.4 z/OS Exemple de chiffrement des journaux actifs du gestionnaire de files d'attente

Les rubriques suivantes vous guident tout au long du processus d'activation du chiffrement des fichiers sur les journaux actifs existants.

Remarque : Le processus des autres fichiers est similaire à celui des journaux actifs.

Dans cet exemple :

- Le gestionnaire de files d'attente CSQ1 est exécuté sous l'utilisateur QMCSQ1et possède des fichiers journaux actifs CSQ1.LOGS.LOGCOPY1.DS001, CSQ1.LOGS.LOGCOPY1.DS002, etc.
- L'environnement matériel et logiciel est capable d'utiliser le chiffrement de fichier z/OS
- RACF est utilisé en tant que fonction d'autorisation système (SAF)
- Le gestionnaire de files d'attente a été arrêté

Effectuez la procédure dans l'ordre suivant:

1. «Configuration de la clé de chiffrement de fichier pour le gestionnaire de files d'attente», à la page 465
2. «Configuration du chiffrement des ensembles de données pour les ensembles de données de journal», à la page 466

V 9.1.4 z/OS Configuration de la clé de chiffrement de fichier pour le gestionnaire de files d'attente

Comment configurer une clé de chiffrement de fichier pour un gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Cette tâche est un prérequis pour «Configuration du chiffrement des ensembles de données pour les ensembles de données de journal», à la page 466.

Procédure

1. Configurez une clé DATA de chiffrement AES-256 bits avec un libellé, par exemple CSQ1DSKY, à l'aide du programme utilitaire du générateur de clés z/OS (KGUP).

2. Définissez le profil RACF CSFKEYS pour la clé de chiffrement CSQ1DSKY en exécutant la commande suivante:

```
RDEFINE CSFKEYS CSQ1DSKY UACC(NONE)
```

3. Configurez le segment ICSF du profil pour permettre l'utilisation de la clé en tant que clé protégée, en exécutant la commande suivante:

```
RALTER CSFKEYS CSQ1DSKY ICSF(SYMCPACFWRAP(YES) SYMCPACFRET(YES))
```

4. Autorisez le gestionnaire de files d'attente à utiliser la clé de chiffrement en donnant à QMCSQ1 un accès en lecture au profil, en exécutant la commande suivante:

```
PERMIT CSQ1DSKY CLASS(CSFKEYS) ID(QMCSQ1) ACCESS(READ)
```

   Accordez le même accès à tout administrateur qui a besoin de lire ou d'écrire le fichier chiffré.

5. Actualisez la classe CSFKEYS en exécutant la commande suivante.

```
SETROPTS RACLIST(CSFKEYS) REFRESH
```

Que faire ensuite

Configurez le chiffrement des ensembles de données pour les ensembles de données comme décrit dans «Configuration du chiffrement des ensembles de données pour les ensembles de données de journal», à la page 466

V 9.1.4 z/OS Configuration du chiffrement des ensembles de données pour les ensembles de données de journal

Comment configurer le chiffrement dans les fichiers journaux.

Avant de commencer

Vérifiez que vous avez lu:

Présentation des étapes de chiffrement d'un fichier IBM MQ for z/OSet exécution de la procédure dans «Configuration de la clé de chiffrement de fichier pour le gestionnaire de files d'attente», à la page 465

Pourquoi et quand exécuter cette tâche

Cette méthode utilise le segment DFP d'un profil générique RACF , afin que vous puissiez utiliser la clé de chiffrement pour tous les nouveaux fichiers qui correspondent au profil.

Vous pouvez également configurer et utiliser une classe de données SMS, ou le libellé de clé peut être spécifié directement lors de l'allocation du fichier.

Comme décrit précédemment, dans cet exemple, le gestionnaire de files d'attente CSQ1 est exécuté sous l'utilisateur QMCSQ1et possède des fichiers journaux actifs CSQ1.LOGS.LOGCOPY1.DS001, CSQ1.LOGS.LOGCOPY1.DS002, etc.

Procédure

1. Créez le profil générique s'il n'existe pas, en exécutant la commande suivante:

```
ADDSD 'CSQ1.LOGS.*' UACC(NONE)
```

2. Autorisez l'utilisateur du gestionnaire de files d'attente à modifier l'accès au profil en exécutant la commande suivante:

```
PERMIT 'CSQ1.LOGS.*' ID(QMCSQ1) ACCESS(ALTER)
```

   De plus, autorisez l'accès approprié nécessaire à tout administrateur.

3. Ajoutez le segment DFP avec le libellé de clé de chiffrement en exécutant la commande suivante:

```
ALTDSD 'CSQ1.LOGS.*' DFP(RESOWNER(QMCSQ1) DATAKEY(CSQ1DSKY))
```

Remarque : Vous devez utiliser la même clé de chiffrement que celle que vous avez utilisée lors de la configuration de la clé de chiffrement de fichier pour le gestionnaire de files d'attente.

4. Actualisez les profils de jeu de données génériques en exécutant la commande suivante:

```
SETROPTS GENERIC(DATASET) REFRESH
```

5. Renommez chaque fichier journal avec une sauvegarde, puis recréez et restaurez les données à l'aide d'IDCAMS. Le fragment JCL suivant convertit CSQ1.LOGS.LOGCOPY1.DS001:

   a) Renommez le jeu de données en un fichier de sauvegarde

```
//RENAME   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RENAME DATASET TO BACKUP                                    */
  /*-------------------------------------------------------------*/
  ALTER 'CSQ1.LOGS.LOGCOPY1.DS001'                  -
        NEWNAME('CSQ1.BAK.LOGS.LOGCOPY1.DS001')
```

   b) Redéfinissez le fichier.

   Le nouveau fichier sera chiffré en raison du profil RACF.

   Remarque : Remplacez + + EXTDCLASS + + par le nom de la classe de données de format étendu que vous souhaitez utiliser pour le fichier.

```
//REDEFINE EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* REDEFINE THE DATASET                                        */
  /*-------------------------------------------------------------*/
  DEFINE CLUSTER                                  -
         (NAME(CSQ1.LOGS.LOGCOPY1.DS001)          -
         LINEAR                                   -
         SHAREOPTIONS(2 3)                        -
         MODEL(CSQ1.BAK.LOGS.LOGCOPY1.DS001)      -
         DATACLAS(++EXTDCLASS++))
```

   c) Copiez les données de la sauvegarde dans le fichier recréé.

   Cette étape chiffre les données:

```
//RESTORE  EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RESTORE DATA INTO ENCRYPTED LOG                             */
  /*-------------------------------------------------------------*/
  REPRO INDATASET(CSQ1.BAK.LOGS.LOGCOPY1.DS001)   -
        OUTDATASET(CSQ1.LOGS.LOGCOPY1.DS001)
```

Que faire ensuite

Répétez l'étape «5», à la page 467 pour tous les fichiers journaux actifs.

Une seule clé de chiffrement est requise et tous les fichiers peuvent être associés au même libellé de clé.

Redémarrez le gestionnaire de files d'attente CSQ1. Utilisez la sortie de la commande DISPLAY LOG pour vérifier que les fichiers journaux ont été chiffrés.

V 9.1.4 z/OS Remarques relatives au chiffrement des fichiers z/OS dans un groupe de partage de files d'attente

Chaque gestionnaire de files d'attente d'un groupe de partage de files d'attente (QSG) doit pouvoir lire les journaux, le fichier d'amorce V 9.1.5 et les fichiers de messages partagés (SMDS), de tous les autres gestionnaires de files d'attente du groupe de partage de files d'attente.

Cela signifie que chaque système sur lequel un membre du groupe de partage de files d'attente peut s'exécuter doit répondre aux exigences de chiffrement de fichier z/OS et que tous les libellés de clé et les clés de chiffrement utilisés pour protéger les fichiers de chaque gestionnaire de files d'attente du groupe de partage de files d'attente doivent être disponibles sur chaque système.

Un gestionnaire de files d'attente antérieur à IBM MQ for z/OS 9.1.3 ne peut pas accéder à un fichier journal actif chiffré.

V 9.1.5 Un gestionnaire de files d'attente antérieur à IBM MQ for z/OS 9.1.3 ne peut pas accéder à un fichier SMDS chiffré.

V 9.1.5 Avant d'utiliser le chiffrement des fichiers z/OS , vous devez migrer tous les gestionnaires de files d'attente d'un groupe de partage de files d'attente vers au moins IBM MQ for z/OS 9.1.3.

Si un gestionnaire de files d'attente d'un groupe de partage de files d'attente est démarré avec un fichier journal actif chiffré et que tout autre gestionnaire de files d'attente du groupe de partage de files d'attente a été démarré, mais qu'il n'a pas été démarré en dernier avec une version de IBM MQ for z/OS prenant en charge les journaux actifs chiffrés, le gestionnaire de files d'attente avec le journal actif chiffré s'arrête de manière anormale avec le code de fin anormale 5C6-00F50033.

V 9.1.5 Vous pouvez convertir un groupe de partage de files d'attente pour utiliser des journaux actifs chiffrés et des fichiers SMDS sans indisponibilité complète, en procédant comme suit:

1. Migration de chaque gestionnaire de files d'attente vers au moins IBM MQ 9.1.5 à tour de rôle.
2. Conversion des journaux actifs en fichiers chiffrés pour chaque gestionnaire de files d'attente. Pour cela, le gestionnaire de files d'attente doit être arrêté, puis redémarré.

   En même temps, il est probable que les ensembles de pages et les journaux d'archivage soient également activés pour les fichiers chiffrés, mais cela n'affecte pas la migration QSG.

   La procédure de conversion de chaque fichier est décrite dans «Exemple de chiffrement des journaux actifs du gestionnaire de files d'attente», à la page 465
3. Pour convertir le fichier SMDS en fichiers chiffrés pour chaque structure d'unité de couplage individuelle, procédez comme suit:
   a. Emission de la commande RESET SMDS (*) ACCESS (DISABLED) CFSTRUCT (structure-name) pour interrompre l'accès du gestionnaire de files d'attente au fichier SMDS.

      Notez que pendant ce temps, les données des files d'attente partagées associées au fichier SMDS sont temporairement indisponibles.
   b. Conversion de chaque fichier qui constitue le fichier SMDS en fichiers chiffrés à l'aide de la procédure décrite dans «Exemple de chiffrement des journaux actifs du gestionnaire de files d'attente», à la page 465.
   c. Emission de la commande RESET SMDS (*) ACCESS (ENABLED) CFSTRUCT (structure-name) pour reprendre l'accès du gestionnaire de files d'attente au fichier SMDS.

Avertissement : Vous devez arrêter correctement le gestionnaire de files d'attente avant de convertir les journaux, et la récupération de la structure d'unité de couplage risque de ne pas être possible lors de la conversion, car les fichiers journaux actifs seront temporairement indisponibles.

V 9.1.4 z/OS Remarques sur la rétromigration lors de l'utilisation du chiffrement des fichiers z/OS

Vous devez prendre en compte les éléments suivants lors de la rétromigration d'un gestionnaire de files d'attente qui possède un ou plusieurs fichiers chiffrés.

Le chiffrement des fichiers z/OS est pris en charge sur les fichiers IBM MQ for z/OS suivants:

- Ensembles de données de journaux actifs
- Ensembles de données de journal d'archivage
- Ensembles de pages
- BSDS
- V 9.1.5 SMDS
- Fichiers CSQINP*

Il n'existe aucune prise en compte de la rétromigration pour les fichiers d'amorce, les journaux d'archivage ou les fichiers CSINP*.

Cependant, il y a des considérations à prendre en compte pour

- V 9.1.5 SMDS
- Ensemble de pages et
- Journal actif

les fichiers, car l'utilisation de ces derniers avec le chiffrement des fichiers z/OS n'est pas prise en charge dans IBM MQ for z/OS 9.1.0et dans les versions antérieures de prise en charge à long terme.

Avant la rétromigration, toutes les règles de chiffrement pour les fichiers SMDS V 9.1.5, les ensembles de pages et les fichiers journaux actifs doivent être supprimées et les données déchiffrées. Ce processus est décrit dans la rubrique «Suppression du chiffrement de fichier d'un fichier», à la page 469.

Avertissement : Si le gestionnaire de files d'attente à rétromirer fait partie d'un groupe de partage de files d'attente (QSG), lisez d'abord la section «Remarques sur les groupes de partage de files d'», à la page 470 .

Suppression du chiffrement de fichier d'un fichier

Cet exemple décrit comment supprimer le chiffrement de fichier du fichier journal CSQ1.LOGS.LOGCOPY1.DS001. Vous pouvez utiliser un processus équivalent pour les ensembles de pages V 9.1.5 SMDS et .

L'exemple suppose que:

- RACF est la fonction d'autorisation système (SAF)
- Le gestionnaire de files d'attente qui utilise le fichier a été arrêté
- Le libellé de la clé de chiffrement a été associé au profil RACF générique CSQ1.LOGS.*

Effectuez la procédure suivante :

1. Copiez les données du fichier dans un fichier de sauvegarde.
   a. Définissez un fichier de sauvegarde qui n'est pas associé à un libellé de clé de chiffrement.

      Remarque : Remplacez + + EXTDCLASS + + par le nom de la classe de données de format étendu que vous souhaitez utiliser pour le fichier.

```
//DEFINE EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
```

```
  /* DEFINE UNENCRYPTED DATA SET                                  */
  /*-------------------------------------------------------------*/
  DEFINE CLUSTER                                  -
         (NAME(CSQ1.BAK.LOGS.LOGCOPY1.DS001)      -
         LINEAR                                   -
         SHAREOPTIONS(2 3)                        -
         MODEL(CSQ1.LOGS.LOGCOPY1.DS001)          -
         DATACLAS(++EXTDCLASS++))
/*
```

b. Copiez les données du jeu de données d'origine dans la sauvegarde. Cette étape déchiffre les données.

```
//COPY     EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* COPY DATA INTO UNENCRYPTED DATA SET                         */
  /*-------------------------------------------------------------*/
  REPRO INDATASET(CSQ1.LOGS.LOGCOPY1.DS001)          -
        OUTDATASET(CSQ1.BAK.LOGS.LOGCOPY1.DS001)
/*
```

c. Supprimer le jeu de données d'origine

```
//DELETE   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* DELETE ORIGINAL                                             */
  /*-------------------------------------------------------------*/
  DELETE ('CSQ1.LOGS.LOGCOPY1.DS001')
/*
```

d. Renommez la sauvegarde avec le nom de fichier d'origine. Les données restent non chiffrées

```
//RENAME   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RENAME UNENCRYPTED DATA SET                                 */
  /*-------------------------------------------------------------*/
  ALTER CSQ1.BAK.LOGS.LOGCOPY1.DS001'               -
         NEWNAME('CSQ1.LOGS.LOGCOPY1.DS001)
  ALTER 'CSQ1.BAK.LOGS.LOGCOPY1.DS001.*'             -
         NEWNAME('CSQ1.LOGS.LOGCOPY1.DS001.*')
/*
```

2. Si vous le souhaitez, répétez ce processus pour les autres fichiers auxquels un libellé de clé de chiffrement est associé via CSQ1.LOGS. * profil générique.
3. (Facultatif) Si tous les fichiers associés à CSQ1.LOGS. * Le profil générique a été déchiffré, supprimez les données DATAKEY associées au profil générique en exécutant la commande suivante:

```
ALTDSD 'CSQ1.LOGS.*' DFP(RESOWNER(QMCSQ1) DATAKEY(CSQ1DSKY))
```

4. Actualisez les profils de jeu de données génériques en exécutant la commande suivante:

```
SETROPTS GENERIC(DATASET) REFRESH
```

5. Redémarrez le gestionnaire de files d'attente.
6. Si la clé de chiffrement n'est plus nécessaire, supprimez-la et supprimez son profil RACF associé de la classe CSFKEYS.

Remarques sur les groupes de partage de files d'

Si un gestionnaire de files d'attente faisant partie d'un groupe de partage de files d'attente doit être rétromigration vers une version de IBM MQ for z/OS qui ne prend pas en charge le chiffrement de fichier, tous les fichiers journaux actifs V 9.1.5 et SMDS de tous les gestionnaires de files d'attente du

groupe de partage de files d'attente doivent être supprimés de leurs règles de chiffrement de fichier et leurs données déchiffrées.

Cela s'applique indépendamment du fait qu'un seul membre du groupe de partage de files d'attente soit rétromigration ou que tous les membres du groupe de partage de files d'attente soient rétromirés.

Vous pouvez supprimer les règles de chiffrement et déchiffrer les données sans indisponibilité complète du groupe de partage de files d'attente:

1. Arrêtez chaque gestionnaire de files d'attente dans le groupe de partage de files d'attente à tour de rôle, en supprimant les règles de chiffrement et en déchiffrant les données de ses journaux actifs, à l'aide du processus décrit dans «Suppression du chiffrement de fichier d'un fichier», à la page 469.

   Si le gestionnaire de files d'attente doit être rétromigration, son ensemble de pages doit également être déchiffré à ce stade. Redémarrez ensuite le gestionnaire de files d'attente.

2. V 9.1.5 Suppression des règles de chiffrement et déchiffrement des données du fichier SMDS de chaque structure d'unité de couplage individuelle en procédant comme suit:

   a. Emission de la commande

   ```
   RESET SMDS(*) ACCESS(DISABLED) CFSTRUCT(structure-name)
   ```

   pour interrompre l'accès du gestionnaire de files d'attente au fichier SMDS. Pendant ce temps, les données des files d'attente partagées associées au fichier SMDS seront temporairement indisponibles.

   b. En suivant le processus dans «Suppression du chiffrement de fichier d'un fichier», à la page 469 pour chaque fichier qui constitue le fichier SMDS.

   c. Emission de la commande

   ```
   RESET SMDS(*) ACCESS(ENABLED) CFSTRUCT(structure-name)
   ```

   pour reprendre l'accès du gestionnaire de files d'attente au fichier SMDS.

Utilisation du chiffrement de fichier z/OS avec un gestionnaire de files d'attente qui ne le prend pas en charge

Si vous effectuez accidentellement une rétromigration d'un gestionnaire de files d'attente vers une version d' IBM MQ for z/OS qui ne prend pas en charge le chiffrement des fichiers et que vous oubliez de supprimer les règles de chiffrement et de déchiffrer les données, une erreur se produit lorsque le gestionnaire de files d'attente tente d'accéder au fichier.

L'erreur dépend du type de fichier et est indiquée dans le tableau suivant.

Remarque : Si une ou plusieurs de ces erreurs se produisent, vous devez suivre les processus décrits dans «Suppression du chiffrement de fichier d'un fichier», à la page 469 pour le fichier concerné. Ces opérations peuvent être effectuées sans changer la version de IBM MQ for z/OS.

Fichier	**Erreur si le gestionnaire de files d'attente ne prend pas en charge le chiffrement des fichiers z/OS**
Ensemble de pages 0	Fin anormale 5C6-00C91400 au démarrage du gestionnaire de files d'attente
Ensembles de pages 1 à 99	MQRC 2193 " Erreur d'ensemble de pages lors de l'accès à l'ensemble de pages, par exemple, sur MQPUT
Journal actif	Fin anormale 5C6-00E80084 au démarrage du gestionnaire de files d'attente
V 9.1.5 SMDS	Le message IEC161I-122 a consigné " Le fichier comporte un élément KEYLABEL, mais l'utilisateur n'a pas indiqué que l'application pouvait gérer le chiffrement.

Fichier	Erreur si le gestionnaire de files d'attente ne prend pas en charge le chiffrement des fichiers z/OS
	SMDS a marqué AVAIL (ERROR).

Intégrité des données de messages

Pour préserver l'intégrité des données, vous pouvez utiliser différents types de programme d'exit utilisateur pour fournir des prétraitements de message ou des signatures numériques pour vos messages.

Intégrité des données

Implémentation de l'intégrité des données dans les messages

Lorsque vous utilisez TLS, votre choix de CipherSpec détermine le niveau d'intégrité des données dans l'entreprise. Si vous utilisez le service AMS (IBM MQ Advanced Message Service), vous pouvez spécifier l'intégrité d'un message unique.

Implémentation de l'intégrité des données dans les exits de message

Un message peut être signé numériquement par un exit de message à l'extrémité émettrice d'un canal. La signature numérique peut alors être vérifiée par une sortie de message à l'extrémité réceptrice d'un canal pour détecter si le message a été volontairement modifié.

Une certaine protection peut être fournie à l'aide d'un résumé de message au lieu d'une signature numérique. Un résumé de message peut être efficace contre les manipulations occasionnelles ou indiscriminées, mais il n'empêche pas l'individu le plus informé de changer ou de remplacer le message, et de générer un résumé complètement nouveau pour lui. Cela est particulièrement vrai si l'algorithme utilisé pour générer le résumé de message est un algorithme bien connu.

Implémentation de l'intégrité des données dans les exits d'envoi et de réception

Sur un canal de message, les exits de message sont plus appropriés pour fournir ce service car un exit de message a accès à l'ensemble d'un message. Sur un canal MQI, les paramètres des appels MQI peuvent contenir des données d'application qui doivent être protégées et seuls les exits d'envoi et de réception peuvent fournir cette protection.

Implémentation de l'intégrité des données dans l'exit API ou l'exit de croisement d'API

Un message peut être signé numériquement par une API ou un exit de croisement d'API lorsque le message est inséré par l'application émettrice. La signature numérique peut alors être vérifiée par une deuxième sortie lorsque le message est récupéré par l'application réceptrice pour détecter si le message a été volontairement modifié.

Une certaine protection peut être fournie à l'aide d'un résumé de message au lieu d'une signature numérique. Un résumé de message peut être efficace contre les manipulations occasionnelles ou indiscriminées, mais il n'empêche pas l'individu le plus informé de changer ou de remplacer le message, et de générer un résumé complètement nouveau pour lui. Ceci est particulièrement vrai si l'algorithme utilisé pour générer le résumé de message est bien connu,

Plus d'informations

Pour plus d'informations sur la garantie de l'intégrité des données, voir la section «Activation des CipherSpecs», à la page 434 .

Tâches associées

Connexion de deux gestionnaires de files d'attente via le protocole TLS

Connexion sécurisée d'un client à un gestionnaire de files d'attente

Audit

Vous pouvez vérifier les intrusions de sécurité ou les tentatives d'intrusion à l'aide de messages d'événement. Vous pouvez également vérifier la sécurité de votre système à l'aide de la IBM MQ Explorer.

Pour détecter les tentatives d'exécution d'actions non autorisées, telles que la connexion à un gestionnaire de files d'attente ou l'insertion d'un message dans une file d'attente, examinez les messages d'événement générés par vos gestionnaires de files d'attente, en particulier les messages d'événement de droits d'accès. Pour plus d'informations sur les messages d'événement du gestionnaire de files d'attente, voir Evénements du gestionnaire de files d'attente, et pour plus d'informations sur la surveillance des événements en général, voir Surveillance des événements.

Maintien de la sécurité des clusters

Autorisez ou empêchez les gestionnaires de files d'attente de rejoindre des clusters ou d'insérer des messages dans des files d'attente de cluster. Forcer un gestionnaire de files d'attente à quitter un cluster. Prenez en compte certaines considérations supplémentaires lors de la configuration de TLS pour les clusters.

Arrêt des gestionnaires de files d'attente non autorisés envoyant des messages

Empêchez les gestionnaires de files d'attente non autorisés d'envoyer des messages à votre gestionnaire de files d'attente à l'aide d'un exit de sécurité de canal.

Avant de commencer

La mise en cluster n'a aucun effet sur le fonctionnement des exits de sécurité. Vous pouvez restreindre l'accès à un gestionnaire de files d'attente de la même manière que dans un environnement de mise en file d'attente répartie.

Pourquoi et quand exécuter cette tâche

Empêchez les gestionnaires de files d'attente sélectionnés d'envoyer des messages à votre gestionnaire de files d'attente:

Procédure

1. Définissez un programme d'exit de sécurité de canal sur la définition de canal CLUSRCVR .
2. Ecrivez un programme qui authentifie les gestionnaires de files d'attente en tentant d'envoyer des messages sur votre canal récepteur de cluster et leur refuse l'accès s'ils ne sont pas autorisés.

Que faire ensuite

Les programmes d'exit de sécurité de canal sont appelés au démarrage et à l'arrêt de l'agent MCA.

Arrêt des gestionnaires de files d'attente non autorisés à insérer des messages dans vos files d'attente

Utilisez l'attribut de droit d'insertion de canal sur le canal récepteur de cluster pour arrêter les gestionnaires de files d'attente non autorisés à placer des messages dans vos files d'attente. Autorisez un gestionnaire de files d'attente éloignées en vérifiant l'ID utilisateur dans le message à l'aide de RACF sur z/OSou de la méthode d'accès aux objets (OAM) sur d'autres plateformes.

Pourquoi et quand exécuter cette tâche

Utilisez les fonctions de sécurité d'une plateforme et le mécanisme de contrôle d'accès dans IBM MQ pour contrôler l'accès aux files d'attente.

Procédure

1. Pour empêcher certains gestionnaires de files d'attente d'insérer des messages dans une file d'attente, utilisez les fonctions de sécurité disponibles sur votre plateforme.

   Exemple :

   - RACF ou autres gestionnaires de sécurité externes sous IBM MQ for z/OS
   - Gestionnaire des droits d'accès aux objets (OAM) sur d'autres plateformes.
2. Utilisez les droits d'insertion, PUTAUT, sur l'attribut de la définition de canal CLUSRCVR .

   L'attribut PUTAUT permet de spécifier les identificateurs utilisateur à utiliser pour établir le droit d'insertion d'un message dans une file d'attente.

   Les options de l'attribut PUTAUT sont les suivantes:

   DEF
   Utilisez l'ID utilisateur par défaut. Sous z/OS, la vérification peut impliquer l'utilisation à la fois de l'ID utilisateur reçu du réseau et de l'ID utilisateur dérivé de MCAUSER.

   CTX
   Utilisez l'ID utilisateur dans les informations de contexte associées au message. Sous z/OS , la vérification peut impliquer l'utilisation de l'ID utilisateur reçu du réseau, ou de l'ID utilisateur dérivé de MCAUSER, ou des deux. Utilisez cette option si le lien est sécurisé et authentifié.

   ONLYMCA (z/OS uniquement)
   Comme pour DEF, mais tout ID utilisateur reçu du réseau n'est pas utilisé. Utilisez cette option si le lien n'est pas sécurisé. Vous souhaitez autoriser uniquement un ensemble spécifique d'actions sur celui-ci, qui sont définies pour MCAUSER.

   ALTMCA (z/OS uniquement)
   Comme pour CTX, mais aucun ID utilisateur reçu du réseau n'est utilisé.

Autorisation d'insertion de messages dans des files d'attente de cluster éloignées

Sur z/OS , configurez l'autorisation d'insertion dans une file d'attente de cluster à l'aide de RACF. Sur les autres plateformes, autorisez l'accès pour la connexion aux gestionnaires de files d'attente et l'insertion dans les files d'attente de ces gestionnaires de files d'attente.

Pourquoi et quand exécuter cette tâche

Le comportement par défaut consiste à effectuer un contrôle d'accès sur le SYSTEM.CLUSTER.TRANSMIT.QUEUE. Notez que ce comportement s'applique, même si vous utilisez plusieurs files d'attente de transmission.

Le comportement spécifique décrit dans cette rubrique s'applique uniquement lorsque vous avez configuré l'attribut **ClusterQueueAccessControl** dans le fichier qm.ini comme étant *RQMName*, comme décrit dans la rubrique Strophe de sécurité , puis redémarré le gestionnaire de files d'attente.

Procédure

- Pour z/OS, exécutez les commandes suivantes:

```
RDEFINE MQQUEUE QMgrName.QUEUE. QueueName UACC(NONE)
PERMIT QMgrName.QUEUE. QueueName CLASS(MQADMIN) ID(GroupName) ACCESS(UPDATE)
```

- Pour les systèmes UNIX, Linux, and Windows , exécutez les commandes suivantes:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

- Pour IBM i, exécutez les commandes suivantes:

```
GRTMQMAUT OBJ(' QMgrName ') OBJTYPE(*MQM) USER(GroupName) AUT(*CONNECT)
GRTMQMAUT OBJ(' QueueName ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

  L'utilisateur peut placer des messages uniquement dans la file d'attente de cluster spécifiée, et aucune autre file d'attente de cluster.

  Les noms de variable ont les significations suivantes:

  QMgrName
  Nom du gestionnaire de files d'attente. Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.

  GroupName
  Nom du groupe auquel l'accès doit être accordé.

  QueueName
  Nom de la file d'attente ou du profil générique pour lequel modifier les autorisations.

Que faire ensuite

Si vous indiquez une file d'attente de réponse lorsque vous placez un message dans une file d'attente de cluster, l'application destinataire doit être autorisée à envoyer la réponse. Définissez ces droits en suivant les instructions de la rubrique «Octroi du droit d'insertion de messages dans une file d'attente de cluster éloignée», à la page 410.

Concepts associés
Section de sécurité dans qm.ini

Empêcher les gestionnaires de files d'attente de rejoindre un cluster

Si un gestionnaire de files d'attente corrompu rejoint un cluster, il est difficile de l'empêcher de recevoir des messages que vous ne souhaitez pas recevoir.

Procédure

Si vous souhaitez vous assurer que seuls certains gestionnaires de files d'attente autorisés rejoignent un cluster, vous avez le choix entre trois techniques:

- A l'aide des enregistrements d'authentification de canal, vous pouvez bloquer la connexion de canal de cluster en fonction de l'adresse IP distante, du nom du gestionnaire de files d'attente éloignées ou du nom distinctif TLS fourni par le système distant.
- Ecrire un programme d'exit pour empêcher les gestionnaires de files d'attente non autorisés d'écrire dans SYSTEM.CLUSTER.COMMAND.QUEUE. Ne limitez pas l'accès à SYSTEM.CLUSTER.COMMAND.QUEUE de sorte qu'aucun gestionnaire de files d'attente ne puisse y écrire, sinon vous empêcheriez tout gestionnaire de files d'attente de rejoindre le cluster.
- Un programme d'exit de sécurité sur la définition de canal CLUSRCVR .

Exits de sécurité sur les canaux de cluster

Remarques supplémentaires à prendre en compte lors de l'utilisation des exits de sécurité sur les canaux de cluster.

Pourquoi et quand exécuter cette tâche

Lorsqu'un canal émetteur de cluster est démarré pour la première fois, il utilise les attributs définis manuellement par un administrateur système. Lorsque le canal est arrêté et redémarré, il récupère les attributs de la définition de canal récepteur de cluster correspondante. La définition de canal émetteur de cluster d'origine est remplacée par les nouveaux attributs, y compris l'attribut SecurityExit .

Procédure

1. Vous devez définir un exit de sécurité à la fois sur l'extrémité émettrice du cluster et sur l'extrémité réceptrice du cluster d'un canal.

   La connexion initiale doit être établie avec un établissement de liaison d'exit de sécurité, même si le nom de l'exit de sécurité est envoyé à partir de la définition du récepteur de cluster.
2. Validez `PartnerName` dans la structure `MQCXP` de l'exit de sécurité.

   L'exit doit autoriser le démarrage du canal uniquement si le gestionnaire de files d'attente partenaire est autorisé
3. Concevez l'exit de sécurité sur la définition de récepteur de cluster à lancer.
4. Si vous le concevez comme étant initié par l'expéditeur, un gestionnaire de files d'attente non autorisé sans exit de sécurité peut rejoindre le cluster car aucun contrôle de sécurité n'est effectué.

   Ce n'est que lorsque le canal est arrêté et redémarré que le nom `SCYEXIT` peut être envoyé à partir de la définition du récepteur de cluster et que des contrôles de sécurité complets ont été effectués.
5. Pour afficher la définition de canal émetteur de cluster en cours d'utilisation, utilisez la commande suivante:

   ```
   DISPLAY CLUSQMGR( queue manager ) ALL
   ```

   La commande affiche les attributs qui ont été envoyés à partir de la définition du récepteur de cluster.
6. Pour afficher la définition d'origine, utilisez la commande suivante:

   ```
   DISPLAY CHANNEL( channel name ) ALL
   ```
7. Vous devrez peut-être définir un exit de définition automatique de canal, `CHADEXIT`, sur le gestionnaire de files d'attente émetteur de cluster, si les gestionnaires de files d'attente se trouvent sur des plateformes différentes.

   Utilisez l'exit de définition automatique de canal pour définir l'attribut `SecurityExit` sur un format approprié pour la plateforme cible.
8. Déployez et configurez l'exit de sécurité.

   z/OS z/OS

   Le module de chargement de l'exit de sécurité doit se trouver dans le fichier spécifié dans l'instruction `CSQXLIB DD` de la procédure d'espace adresse de l'initiateur de canal.

   ULW Windows, systèmes UNIX and Linux

   - La bibliothèque de liens dynamiques d'exit de sécurité doit se trouver dans le chemin indiqué dans l'attribut `SCYEXIT` de la définition de canal.
   - La bibliothèque de liaison dynamique de l'exit de définition automatique de canal doit se trouver dans le chemin indiqué dans l'attribut `CHADEXIT` de la définition de gestionnaire de files d'attente.

Forcer les gestionnaires de files d'attente indésirables à quitter un cluster

Forcez un gestionnaire de files d'attente non souhaité à quitter un cluster en exécutant la commande `RESET CLUSTER` sur un gestionnaire de files d'attente de référentiel complet.

Pourquoi et quand exécuter cette tâche

Vous pouvez forcer un gestionnaire de files d'attente indésirable à quitter un cluster. Si, par exemple, un gestionnaire de files d'attente est supprimé mais que ses canaux récepteurs de cluster sont toujours définis dans le cluster. Vous voudrez peut-être ranger.

Seuls les gestionnaires de files d'attente de référentiel complet sont autorisés à éjecter un gestionnaire de files d'attente d'un cluster.

Remarque : Bien que l'utilisation de la commande RESET CLUSTER force la suppression d'un gestionnaire de files d'attente d'un cluster, l'utilisation de la commande RESET CLUSTER seule n'empêche pas le gestionnaire de files d'attente de rejoindre le cluster ultérieurement. Pour vous assurer que le gestionnaire de files d'attente ne rejoint pas le cluster, suivez les étapes décrites dans «Empêcher les gestionnaires de files d'attente de rejoindre un cluster», à la page 475.

Procédez comme suit pour éjecter le gestionnaire de files d'attente OSLO du cluster NORWAY:

Procédure

1. Sur un gestionnaire de files d'attente de référentiel complet, exécutez la commande suivante:

```
RESET CLUSTER(NORWAY) QMNAME(OSLO) ACTION(FORCEREMOVE)
```

2. Vous pouvez également utiliser QMID à la place de QMNAME dans la commande:

```
RESET CLUSTER(NORWAY) QMID(qmid) ACTION(FORCEREMOVE)
```

   Remarque : QMID est une chaîne. Par conséquent, la valeur de qmid doit être placée entre apostrophes, par exemple, QMID('FR01_2019-07-15_14.42.42').

Résultats

Le gestionnaire de files d'attente qui est supprimé de force ne change pas ; ses définitions de cluster local indiquent qu'il se trouve dans le cluster. Les définitions de tous les autres gestionnaires de files d'attente ne l'affichent pas dans le cluster.

Empêcher les gestionnaires de files d'attente de recevoir des messages

Vous pouvez empêcher un gestionnaire de files d'attente de cluster de recevoir des messages qu'il n'est pas autorisé à recevoir à l'aide de programmes d'exit.

Pourquoi et quand exécuter cette tâche

Il est difficile d'empêcher un gestionnaire de files d'attente membre d'un cluster de définir une file d'attente. Il existe un risque qu'un gestionnaire de files d'attente incontrôlable rejoigne un cluster et définisse sa propre instance de l'une des files d'attente du cluster. Il peut désormais recevoir des messages qu'il n'est pas autorisé à recevoir. Pour empêcher un gestionnaire de files d'attente de recevoir des messages, utilisez l'une des options suivantes fournies dans la procédure.

Procédure

- Un programme d'exit de canal sur chaque canal émetteur de cluster. Le programme d'exit utilise le nom de connexion pour déterminer si le gestionnaire de files d'attente de destination est approprié pour l'envoi des messages.
- Un programme d'exit de charge de travail de cluster, qui utilise les enregistrements de destination pour déterminer l'adéquation de la file d'attente de destination et du gestionnaire de files d'attente pour l'envoi des messages.

SSL/TLS et clusters

Lors de la configuration de TLS pour les clusters, sachez qu'une définition de canal CLUSRCVR est propagée à d'autres gestionnaires de files d'attente en tant que canal CLUSSDR défini automatiquement. Si un canal CLUSRCVR utilise TLS, vous devez configurer TLS sur tous les gestionnaires de files d'attente qui communiquent à l'aide du canal.

Pour plus d'informations sur le protocole TLS, voir «Protocoles de sécurité TLS dans IBM MQ», à la page 24. Les conseils qui s'y appliquent sont généralement applicables aux canaux de cluster, mais vous souhaiterez peut-être accorder une attention particulière aux éléments suivants:

Dans un cluster IBM MQ , une définition de canal CLUSRCVR particulière est fréquemment propagée à de nombreux autres gestionnaires de files d'attente où elle est transformée en un CLUSSDRdéfini automatiquement. Par la suite, le CLUSSDR défini automatiquement est utilisé pour démarrer un canal vers CLUSRCVR. Si CLUSRCVR est configuré pour la connectivité TLS, les considérations suivantes s'appliquent:

- Tous les gestionnaires de files d'attente qui souhaitent communiquer avec ce CLUSRCVR doivent avoir accès au support TLS. Cette mise à disposition TLS doit prendre en charge le CipherSpec pour le canal.
- Les différents gestionnaires de files d'attente auxquels les canaux émetteurs de cluster définis automatiquement ont été propagés auront chacun un nom distinctif différent associé. Si la vérification d'homologue de nom distinctif doit être utilisée sur le CLUSRCVR , elle doit être configurée de sorte que tous les noms distinctifs pouvant être reçus soient correctement mis en correspondance.

  Par exemple, supposons que tous les gestionnaires de files d'attente qui hébergeront des canaux émetteurs de cluster qui se connecteront à un CLUSRCVRparticulier soient associés à des certificats. Supposons également que les noms distinctifs de tous ces certificats définissent le pays en tant que Royaume-Uni, l'organisation en tant que IBM, l'unité organisationnelle en tant que IBM MQ Development, et aient tous des noms communs sous la forme DEVT.QMnnn, où nnn est numérique.

  Dans ce cas, la valeur SSLPEER de `C=UK, O=IBM, OU=IBM MQ Development, CN=DEVT.QM*` sur le CLUSRCVR permet à tous les canaux émetteurs de cluster requis de se connecter correctement, mais empêche les canaux émetteurs de cluster indésirables de se connecter.
- Si des chaînes CipherSpec personnalisées sont utilisées, sachez que les formats de chaîne personnalisés ne sont pas autorisés sur toutes les plateformes. Par exemple, la chaîneCipherSpec RC4_SHA_US a la valeur 05 on IBM i mais n'est pas une spécification valide sur les systèmes UNIX, Linux ou Windows . Ainsi, si des paramètres SSLCIPH personnalisés sont utilisés sur un CLUSRCVR, tous les canaux émetteurs de cluster définis automatiquement doivent résider sur des plateformes sur lesquelles le support TLS sous-jacent implémente ce CipherSpec et sur lesquelles il peut être spécifié avec la valeur personnalisée. Si vous ne pouvez pas sélectionner une valeur pour le paramètre SSLCIPH qui sera comprise dans votre cluster, vous aurez besoin d'un exit de définition automatique de canal pour que les plateformes utilisées puissent la comprendre. Utilisez les chaînes textuelles CipherSpec lorsque cela est possible (par exemple, TLS_RSA_WITH_AES_128_CBC_SHA).

Un paramètre SSLCRLNL s'applique à un gestionnaire de files d'attente individuel et n'est pas propagé à d'autres gestionnaires de files d'attente au sein d'un cluster.

Mise à niveau des gestionnaires de files d'attente en cluster et des canaux vers SSL/TLS

Mettez à niveau les canaux de cluster un par un, en modifiant tous les canaux CLUSRCVR avant les canaux CLUSSDR .

Avant de commencer

Tenez compte des considérations suivantes, car elles peuvent affecter votre choix de CipherSpec pour un cluster:

- Certains CipherSpecs ne sont pas disponibles sur toutes les plateformes. Prenez soin de choisir un CipherSpec pris en charge par tous les gestionnaires de files d'attente du cluster.
- Certains CipherSpecs peuvent être nouveaux dans la version actuelle de IBM MQ et ne pas être pris en charge dans les versions plus anciennes. Un cluster contenant des gestionnaires de files d'attente s'exécutant dans différentes éditions de MQ ne peut utiliser que les CipherSpecs prises en charge par chaque édition.

  Pour utiliser un nouveau CipherSpec dans un cluster, vous devez d'abord migrer tous les gestionnaires de files d'attente de cluster vers l'édition en cours.

- Certains CipherSpecs nécessitent l'utilisation d'un type spécifique de certificat numérique, notamment ceux qui utilisent la cryptographie Elliptic Curve.

Avertissement : Il n'est pas possible d'utiliser un mélange de certificats signés par Elliptic Curve et de certificats signés par RSA sur les gestionnaires de files d'attente que vous souhaitez joindre dans le cadre d'un cluster.

Les gestionnaires de files d'attente d'un cluster doivent tous utiliser des certificats signés par RSA ou tous utiliser des certificats signés par EC, et non une combinaison des deux.

Pour plus d'informations, voir «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45.

Mettez à niveau tous les gestionnaires de files d'attente du cluster vers IBM MQ V8 ou ultérieure, s'ils ne sont pas déjà à ces niveaux. Distribuez les certificats et les clés pour que TLS fonctionne à partir de chacun d'eux.

Si vous souhaitez mettre à niveau Tom ou utiliser les CipherSpecs ANY_TLS12 , vous devez mettre à niveau tous les gestionnaires de files d'attente du cluster vers IBM MQ 9.1.2 ou une version ultérieure.

Si vous souhaitez effectuer une mise à niveau ou utiliser l'un des autres alias CipherSpecs (ANY_TLS13, ANY_TLS12, ANY_TLS12_OR_HIGHER, etc.), vous devez mettre à niveau tous les gestionnaires de files d'attente du cluster vers IBM MQ 9.1.4 ou une version ultérieure.

Pourquoi et quand exécuter cette tâche

Modifiez les canaux CLUSRCVR avant les canaux CLUSSDR .

Procédure

1. Basculez les canaux CLUSRCVR vers TLS dans l'ordre de votre choix, en modifiant un CLUSRCVR à la fois, et autorisez les modifications à circuler dans le cluster avant de modifier le suivant.

   Important : Veillez à ne pas modifier le chemin inverse tant que les modifications du canal en cours n'ont pas été distribuées dans le cluster.
2. Facultatif : Commuter tous les canaux CLUSSDR manuels vers TLS.

   Cela n'a aucun effet sur le fonctionnement du cluster, sauf si vous utilisez la commande REFRESH CLUSTER avec l'option REPOS(YES) .

   Remarque : Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster pendant qu'il est en cours, et à nouveau à des intervalles de 27 jours par la suite lorsque les objets de cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster.
3. Utilisez la commande DISPLAY CLUSQMGR pour vous assurer que la nouvelle configuration de sécurité a été propagée dans le cluster.
4. Redémarrez les canaux pour utiliser TLS et exécutez REFRESH SECURITY (SSL).

Concepts associés

«Activation des CipherSpecs», à la page 434
Activez un CipherSpec à l'aide du paramètre **SSLCIPH** dans la commande **DEFINE CHANNEL** MQSC ou dans la commande **ALTER CHANNEL** MQSC.

«Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45
Cette rubrique fournit des informations sur la façon de choisir les CipherSpecs et les certificats numériques appropriés pour votre règle de sécurité, en soulignant la relation entre les CipherSpecs et les certificats numériques dans IBM MQ.

Information associée

Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER

Désactivation de SSL/TLS sur les gestionnaires de files d'attente et les canaux en cluster

Pour désactiver TLS, définissez le paramètre SSLCIPH sur ' '. Désactivez TLS sur les canaux de cluster individuellement, en modifiant tous les canaux récepteurs de cluster avant les canaux émetteurs de cluster.

Pourquoi et quand exécuter cette tâche

Modifiez un canal récepteur de cluster à la fois et autorisez les modifications à transiter par le cluster avant de modifier le suivant.

Important : Veillez à ne pas modifier le chemin inverse tant que les modifications du canal en cours n'ont pas été distribuées dans le cluster.

Procédure

1. Définissez la valeur du paramètre SSLCIPH sur ' ', une chaîne vide entre apostrophes ou *NONE sur IBM i .

   Vous pouvez désactiver TLS sur les canaux récepteurs de cluster dans l'ordre de votre choix.

   Notez que les modifications circulent dans la direction opposée sur les canaux sur lesquels vous laissez TLS actif.
2. Vérifiez que la nouvelle valeur est reflétée dans tous les autres gestionnaires de files d'attente à l'aide de la commande **DISPLAY CLUSQMGR(*)** ALL.
3. Désactivez TLS sur tous les canaux émetteurs de cluster manuels.

   Cela n'a aucun effet sur le fonctionnement du cluster, sauf si vous utilisez la commande **REFRESH CLUSTER** avec l'option REPOS (YES).

   Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster pendant qu'il est en cours, puis à intervalles réguliers, lorsque les objets de cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Pour plus d'informations, voir La régénération dans un cluster de grande taille peut affecter les performances et la disponibilité du cluster .
4. Arrêtez et redémarrez les canaux émetteurs de cluster.

Sécurité de publication / abonnement

Les composants et les interactions impliqués dans la publication / l'abonnement sont décrits comme une introduction aux explications et exemples plus détaillés qui suivent.

Un certain nombre de composants sont impliqués dans la publication et l'abonnement à une rubrique. Certaines des relations de sécurité entre eux sont illustrées dans Figure 22, à la page 481 et décrites dans l'exemple suivant.

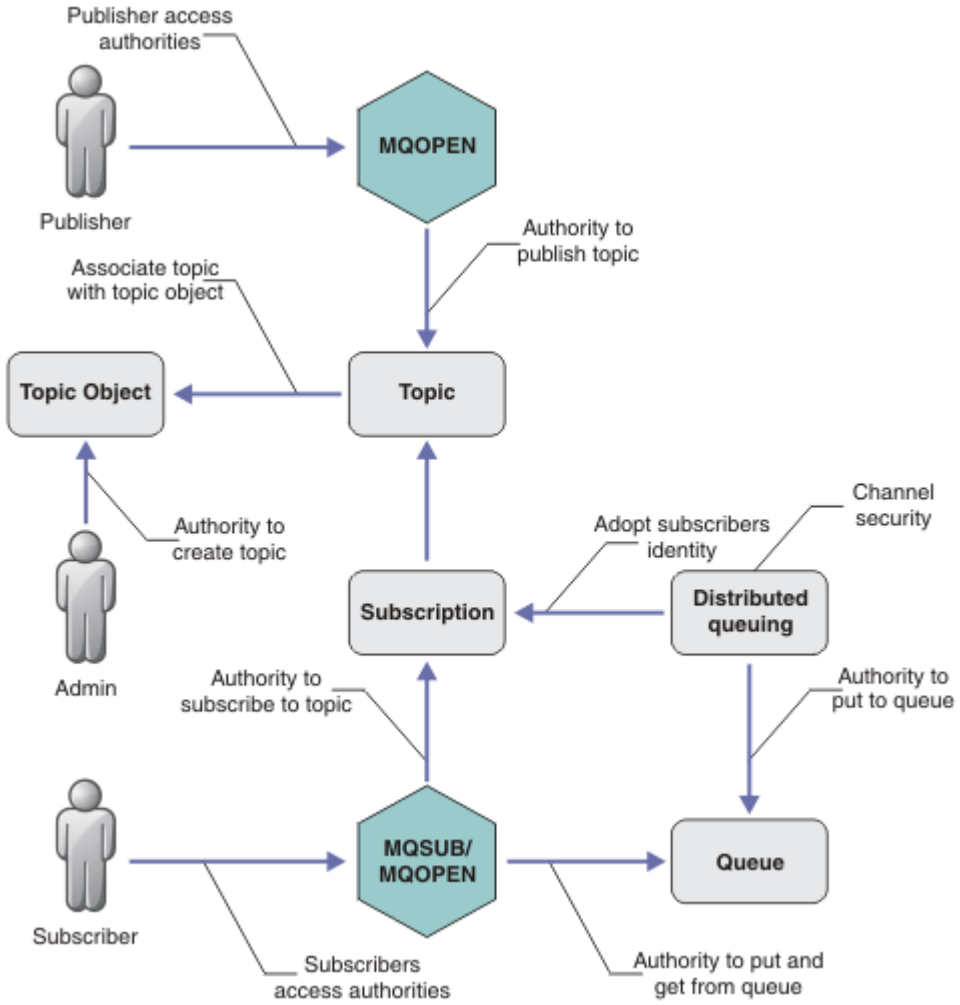


Figure 22. Relations de sécurité de publication / abonnement entre différents composants

Rubriques

Les rubriques sont identifiées par des chaînes de rubrique et sont généralement organisées en arborescences. Voir Arborescences de rubriques. Vous devez associer une rubrique à un objet de rubrique pour contrôler l'accès à la rubrique. «Modèle de sécurité de rubrique», à la page 483 explique comment sécuriser des rubriques à l'aide d'objets de rubrique.

Objets de rubrique d'administration

Vous pouvez contrôler qui a accès à une rubrique et dans quel but, à l'aide de la commande **setmqaut** avec une liste d'objets de rubrique d'administration. Consultez les exemples, «Accorder l'accès à un utilisateur pour s'abonner à une rubrique», à la page 488 et «Accorder l'accès à un utilisateur pour la publication dans une rubrique», à la page 496. z/OS Pour contrôler l'accès aux objets de rubrique sur z/OS, voir Profils pour la sécurité des rubriques.

Abonnements

Abonnez-vous à une ou plusieurs rubriques en créant un abonnement fournissant une chaîne de rubrique, qui peut inclure des caractères génériques, à mettre en correspondance avec les chaînes de rubrique des publications. Pour plus de détails, voir:

S'abonner à l'aide d'un objet de rubrique

«Abonnement à l'aide du nom d'objet de rubrique», à la page 484

S'abonner à l'aide d'une rubrique
«Abonnement à l'aide d'une chaîne de rubrique dans laquelle le noeud de rubrique n'existe pas», à la page 485

S'abonner à l'aide d'une rubrique avec des caractères génériques
«Abonnement à l'aide d'une chaîne de sujet contenant des caractères génériques», à la page 486

Un abonnement contient des informations sur l'identité de l'abonné et sur l'identité de la file d'attente de destination dans laquelle les publications doivent être placées. Il contient également des informations sur la manière dont la publication doit être placée dans la file d'attente de destination.

En plus de définir les abonnés qui ont le droit de s'abonner à certaines rubriques, vous pouvez limiter les abonnements à l'utilisation par un abonné individuel. Vous pouvez également contrôler les informations sur l'abonné qui sont utilisées par le gestionnaire de files d'attente lorsque des publications sont placées dans la file d'attente de destination. Voir «Sécurité des abonnements», à la page 501.

Files d'attente

La file d'attente de destination est une file d'attente importante à sécuriser. Il est local pour l'abonné et les publications qui correspondent à l'abonnement y sont placées. Vous devez envisager d'accéder à la file d'attente de destination à partir de deux perspectives:

1. Insertion d'une publication dans la file d'attente de destination.
2. Extraction de la publication de la file d'attente de destination.

Le gestionnaire de files d'attente place une publication dans la file d'attente de destination à l'aide d'une identité fournie par l'abonné. L'abonné, ou un programme auquel la tâche d'obtention de publications a été déléguée, enlève les messages de la file d'attente. Voir «Droits d'accès aux files d'attente de destination», à la page 486.

Il n'existe pas d'alias d'objet de rubrique, mais vous pouvez utiliser une file d'attente alias comme alias d'un objet de rubrique. Dans ce cas, en plus de vérifier les droits d'utilisation de la rubrique pour la publication ou l'abonnement, le gestionnaire de files d'attente vérifie les droits d'utilisation de la file d'attente.

«Sécurité de publication / abonnement entre les gestionnaires de files d'attente», à la page 503

Votre droit de publication ou d'abonnement à une rubrique est vérifié sur le gestionnaire de files d'attente local à l'aide des identités et des autorisations locales. L'autorisation ne dépend pas de la définition ou non de la rubrique, ni de l'endroit où elle est définie. Par conséquent, vous devez effectuer une autorisation de rubrique sur chaque gestionnaire de files d'attente d'un cluster lorsque des rubriques en cluster sont utilisées.

Remarque : Le modèle de sécurité des rubriques diffère du modèle de sécurité des files d'attente. Vous pouvez obtenir le même résultat pour les files d'attente en définissant un alias de file d'attente en local pour chaque file d'attente en cluster.

Les gestionnaires de files d'attente échangent des abonnements dans un cluster. Dans la plupart des configurations de cluster IBM MQ , les canaux sont configurés avec PUTAUT=DEF pour placer des messages dans des files d'attente cible en utilisant les droits du processus de canal. Vous pouvez modifier la configuration de canal pour utiliser PUTAUT=CTX afin que l'utilisateur abonné ait le droit de propager un abonnement à un autre gestionnaire de files d'attente dans un cluster.

«Sécurité de publication / abonnement entre les gestionnaires de files d'attente», à la page 503 décrit comment modifier vos définitions de canal pour contrôler qui est autorisé à propager des abonnements sur d'autres serveurs du cluster.

Authorization

Vous pouvez appliquer une autorisation à des objets de rubrique, tout comme des files d'attente et d'autres objets. Il existe trois opérations d'autorisation, pub, subet `resume` , que vous pouvez appliquer uniquement aux rubriques. Les détails sont décrits dans Spécification des droits pour différents types d'objet.

Appels de fonction

Dans les programmes de publication et d'abonnement, comme dans les programmes en file d'attente, des vérifications d'autorisation sont effectuées lorsque des objets sont ouverts, créés, modifiés ou supprimés. Les vérifications ne sont pas effectuées lorsque des appels MQPUT ou MQGET MQI sont effectués pour placer et obtenir des publications.

Pour publier une rubrique, effectuez un MQOPEN sur la rubrique, qui effectue les vérifications d'autorisation. Publiez des messages dans le descripteur de rubrique à l'aide de la commande MQPUT , qui n'effectue aucune vérification d'autorisation.

Pour vous abonner à une rubrique, vous exécutez généralement une commande MQSUB pour créer ou reprendre l'abonnement, ainsi que pour ouvrir la file d'attente de destination afin de recevoir des publications. Vous pouvez également effectuer un MQOPEN distinct pour ouvrir la file d'attente de destination, puis exécuter la commande MQSUB pour créer ou reprendre l'abonnement.

Quels que soient les appels que vous utilisez, le gestionnaire de files d'attente vérifie que vous pouvez vous abonner à la rubrique et obtenir les publications résultantes de la file d'attente de destination. Si la file d'attente de destination n'est pas gérée, des vérifications d'autorisation sont également effectuées pour que le gestionnaire de files d'attente puisse placer des publications dans la file d'attente de destination. Il utilise l'identité qu'il a adoptée à partir d'un abonnement correspondant. Il est supposé que le gestionnaire de files d'attente est toujours en mesure de placer des publications dans des files d'attente de destination gérées.

Rôles

Les utilisateurs sont impliqués dans quatre rôles lors de l'exécution d'applications de publication / abonnement:

1. Diffuseur de publications
2. Abonné
3. Administrateur de rubriques
4. IBM MQ Administrateur-membre du groupe mqm

Définissez des groupes avec les autorisations appropriées correspondant aux rôles de publication, d'abonnement et d'administration de sujet. Vous pouvez ensuite affecter des principaux à ces groupes en les autorisant à effectuer des tâches de publication et d'abonnement spécifiques.

En outre, vous devez étendre les autorisations d'opérations d'administration à l'administrateur des files d'attente et des canaux en charge du déplacement des publications et des abonnements.

Modèle de sécurité de rubrique

Seuls les objets de rubrique définis peuvent être associés à des attributs de sécurité. Pour obtenir une description des objets de rubrique, voir Objets de rubrique d'administration. Les attributs de sécurité indiquent si un ID utilisateur ou un groupe de sécurité spécifié est autorisé à effectuer une opération d'abonnement ou de publication sur chaque objet de rubrique.

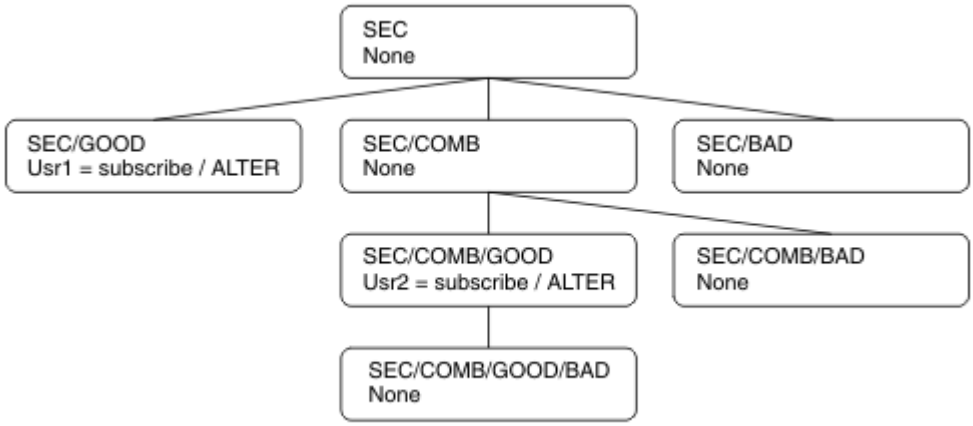
Les attributs de sécurité sont associés au noeud d'administration approprié dans l'arborescence de rubriques. Lorsqu'une vérification des droits est effectuée pour un ID utilisateur particulier lors d'une opération d'abonnement ou de publication, les droits accordés sont basés sur les attributs de sécurité du noeud d'arborescence de rubriques associé.

Les attributs de sécurité sont une liste de contrôle d'accès qui indique les droits d'accès d'un ID utilisateur ou d'un groupe de sécurité du système d'exploitation sur l'objet de rubrique.

Prenez l'exemple suivant dans lequel les objets de rubrique ont été définis avec les attributs de sécurité ou les droits affichés:

Tableau 80. Exemples de droits sur les objets de rubrique			
Nom de la rubrique	**Chaîne de rubrique**	**Droits d'accès-non z/OS**	**z/OS droits**
SECROOT	SEC	Aucun	Aucun
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD
SECBAD	SEC/BAD	Aucun	Aucun HLQ.SUBSCRIBE.SECBAD
SECCOMB	SEC/COMB	Aucun	Aucun HLQ.SUBSCRIBE.SECCOMB
SECCOMBB	SEC/COMB/ GOOD/BAD	Aucun	Aucun HLQ.SUBSCRIBE.SECCOMBB
SECCOMBG	SEC/COMB/GOOD	usr2+subscribe	ALTER HLQ.SUBSCRIBE.SECCOMBG
SECCOMBN	SEC/COMB/BAD	Aucun	Aucun HLQ.SUBSCRIBE.SECCOMBN

L'arborescence de rubriques avec les attributs de sécurité associés sur chaque noeud peut être représentée comme suit:



Les exemples répertoriés donnent les autorisations suivantes:

- Sur le noeud racine de l'arborescence /SEC, aucun utilisateur n'a de droits sur ce noeud.
- usr1 a reçu le droit d'abonnement à l'objet /SEC/GOOD
- usr2 a reçu le droit d'abonnement à l'objet /SEC/COMB/GOOD

Abonnement à l'aide du nom d'objet de rubrique

Lors de l'abonnement à un objet de rubrique en spécifiant le nom MQCHAR48 , le noeud correspondant dans l'arborescence de rubriques est localisé. Si les attributs de sécurité associés au noeud indiquent que l'utilisateur est autorisé à s'abonner, l'accès est accordé.

Si l'accès n'est pas accordé à l'utilisateur, le noeud parent de l'arborescence détermine si l'utilisateur est autorisé à s'abonner au niveau du noeud parent. Si tel est le cas, l'accès est accordé. Si ce n'est pas le cas,

le parent de ce noeud est pris en compte. La récursivité se poursuit jusqu'à ce qu'un noeud soit localisé et accorde le droit d'abonnement à l'utilisateur. La récursivité s'arrête lorsque le noeud racine est pris en compte sans que les droits aient été accordés. Dans ce dernier cas, l'accès est refusé.

En résumé, si un noeud du chemin accorde le droit de s'abonner à cet utilisateur ou à cette application, l'abonné est autorisé à s'abonner à ce noeud ou à n'importe quel emplacement situé en dessous de ce noeud dans l'arborescence de rubriques.

Le noeud racine de l'exemple est SEC.

Le droit d'abonnement est accordé à l'utilisateur si la liste de contrôle d'accès indique que l'ID utilisateur lui-même dispose de droits ou qu'un groupe de sécurité du système d'exploitation dont l'ID utilisateur est membre dispose de droits.

Ainsi, par exemple:

- Si `usr1` tente de s'abonner à l'aide d'une chaîne de rubrique `SEC/GOOD`, l'abonnement est autorisé car l'ID utilisateur a accès au noeud associé à cette rubrique. Toutefois, si `usr1` tente de s'abonner à l'aide de la chaîne de rubrique `SEC/COMB/GOOD` , l'abonnement ne sera pas autorisé car l'ID utilisateur n'a pas accès au noeud qui lui est associé.
- Si `usr2` tente de s'abonner, à l'aide d'une chaîne de rubrique `SEC/COMB/GOOD` , l'abonnement est autorisé car l'ID utilisateur a accès au noeud associé à la rubrique. Toutefois, si `usr2` tentait de s'abonner à `SEC/GOOD` , l'abonnement ne serait pas autorisé car l'ID utilisateur n'a pas accès au noeud qui lui est associé.
- Si `usr2` tente de s'abonner à l'aide d'une chaîne de rubrique `SEC/COMB/GOOD/BAD` , l'abonnement est autorisé car l'ID utilisateur a accès au noeud parent `SEC/COMB/GOOD`.
- Si `usr1` ou `usr2` tente de s'abonner à l'aide d'une chaîne de rubrique `/SEC/COMB/BAD`, aucune n'est autorisée car ils n'ont pas accès au noeud de rubrique qui lui est associé, ni aux noeuds parent de cette rubrique.

Une opération d'abonnement spécifiant le nom d'un objet de rubrique qui n'existe pas génère une erreur MQRC_UNKNOWN_OBJECT_NAME.

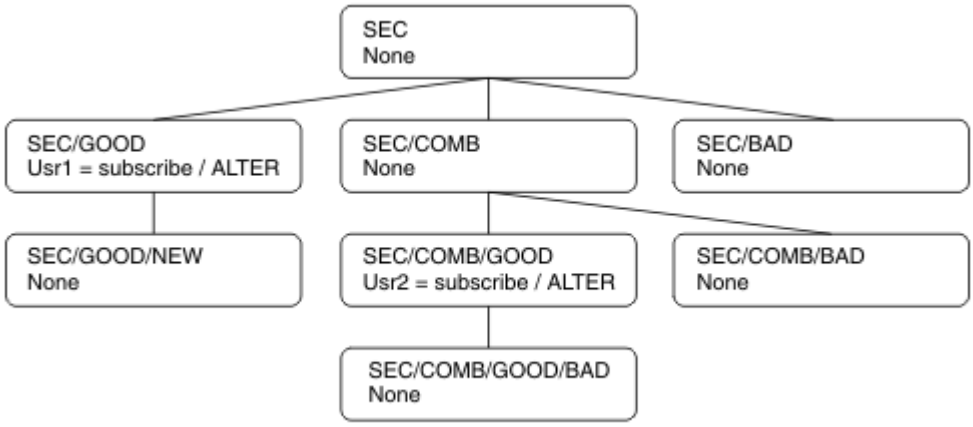
Abonnement à l'aide d'une chaîne de rubrique dans laquelle le noeud de rubrique existe

Le comportement est le même que lors de la spécification de la rubrique par le nom d'objet MQCHAR48 .

Abonnement à l'aide d'une chaîne de rubrique dans laquelle le noeud de rubrique n'existe pas

Prenons le cas d'une application abonnée, en spécifiant une chaîne de rubrique représentant un noeud de rubrique qui n'existe pas actuellement dans l'arborescence de rubriques. La vérification des droits d'accès est effectuée comme indiqué dans la section précédente. La vérification commence par le noeud parent de celui qui est représenté par la chaîne de rubrique. Si les droits sont accordés, un nouveau noeud représentant la chaîne de rubrique est créé dans l'arborescence de rubriques.

Par exemple, `usr1` tente de s'abonner à une rubrique `SEC/GOOD/NEW`. Les droits sont accordés car `usr1` a accès au noeud parent `SEC/GOOD`. Un nouveau noeud de rubrique est créé dans l'arborescence, comme le montre le diagramme suivant. Le nouveau noeud de rubrique n'est pas un objet de rubrique auquel aucun attribut de sécurité n'est directement associé ; les attributs sont hérités de son parent.



Abonnement à l'aide d'une chaîne de sujet contenant des caractères génériques

Prenez en compte le cas de l'abonnement à l'aide d'une chaîne de rubrique contenant un caractère générique. La vérification des droits est réalisée sur le noeud dans l'arborescence de sujets qui correspond à la partie qualifiée complète de la chaîne de sujet.

Par conséquent, si une application s'abonne à SEC/COMB/GOOD/*, une vérification des droits d'accès est effectuée comme indiqué dans les deux sections précédentes sur le noeud SEC/COMB/GOOD dans l'arborescence de rubriques.

De même, si une application doit s'abonner à SEC/COMB/*/GOOD, une vérification des droits d'accès est effectuée sur le noeud SEC/COMB.

Droits d'accès aux files d'attente de destination

Lors de l'abonnement à une rubrique, l'un des paramètres est le descripteur hobj d'une file d'attente qui a été ouverte pour la sortie afin de recevoir les publications.

Si hobj n'est pas spécifié, mais qu'il est vide, une file d'attente gérée est créée si les conditions suivantes s'appliquent:

- L'option MQSO_MANAGED a été spécifiée.
- L'abonnement n'existe pas.
- La création est spécifiée.

Si hobj est vide et que vous modifiez ou reprenez un abonnement existant, la file d'attente de destination précédemment fournie peut être gérée ou non gérée.

L'application ou l'utilisateur qui effectue la demande MQSUB doit avoir le droit d'insérer des messages dans la file d'attente de destination qu'elle a fournie ; en effet, il doit avoir le droit d'insérer des messages publiés dans cette file d'attente. La vérification des droits d'accès suit les règles existantes pour la vérification de la sécurité de la file d'attente.

La vérification de la sécurité inclut un ID utilisateur alternatif et des vérifications de la sécurité du contexte, le cas échéant. Pour pouvoir définir l'une des zones de contexte d'identité, vous devez spécifier l'option MQSO_SET_IDENTITY_CONTEXT ainsi que l'option MQSO_CREATE ou MQSO_ALTER . Vous ne pouvez pas définir de zones de contexte d'identité dans une demande MQSO_RESUME .

Si la destination est une file d'attente gérée, aucun contrôle de sécurité n'est effectué sur la destination gérée. Si vous êtes autorisé à vous abonner à une rubrique, il est supposé que vous pouvez utiliser des destinations gérées.

Publication à l'aide du nom de rubrique ou de la chaîne de rubrique dans laquelle le noeud de rubrique existe

Le modèle de sécurité pour la publication est le même que pour l'abonnement, à l'exception des caractères génériques. Les publications ne contiennent pas de caractères génériques ; il n'y a donc pas de cas d'une chaîne de rubrique contenant des caractères génériques à prendre en compte.

Les droits de publication et d'abonnement sont distincts. Un utilisateur ou un groupe peut avoir le droit d'en effectuer un sans être nécessairement en mesure d'en effectuer un autre.

Lors de la publication dans un objet de rubrique en spécifiant le nom MQCHAR48 ou la chaîne de rubrique, le noeud correspondant dans l'arborescence de rubriques est localisé. Si les attributs de sécurité associés au noeud de rubrique indiquent que l'utilisateur est autorisé à publier, l'accès est accordé.

Si l'accès n'est pas accordé, le noeud parent de l'arborescence détermine si l'utilisateur a le droit de publier à ce niveau. Si tel est le cas, l'accès est accordé. Si ce n'est pas le cas, la récursivité se poursuit jusqu'à ce qu'un noeud soit localisé et accorde le droit de publication à l'utilisateur. La récursivité s'arrête lorsque le noeud racine est pris en compte sans que les droits aient été accordés. Dans ce dernier cas, l'accès est refusé.

En bref, si un noeud du chemin accorde le droit de publier à cet utilisateur ou à cette application, le diffuseur de publications est autorisé à publier sur ce noeud ou n'importe où en dessous de ce noeud dans l'arborescence de rubriques.

Publication à l'aide du nom de rubrique ou de la chaîne de rubrique où le noeud de rubrique n'existe pas

Comme pour l'opération d'abonnement, lorsqu'une application publie, en spécifiant une chaîne de rubrique représentant un noeud de rubrique qui n'existe pas actuellement dans l'arborescence de rubriques, la vérification des droits d'accès est effectuée en commençant par le parent du noeud représenté par la chaîne de rubrique. Si les droits sont accordés, un nouveau noeud représentant la chaîne de rubrique est créé dans l'arborescence de rubriques.

Publication à l'aide d'une file d'attente alias qui se résout en un objet de rubrique

Si vous publiez à l'aide d'une file d'attente alias qui se résout en un objet de rubrique, la vérification de la sécurité est effectuée à la fois sur la file d'attente alias et sur la rubrique sous-jacente à laquelle elle se résout.

Le contrôle de sécurité de la file d'attente alias vérifie que l'utilisateur est autorisé à placer des messages dans cette file d'attente alias et le contrôle de sécurité de la rubrique vérifie que l'utilisateur peut publier des messages dans cette rubrique. Lorsqu'une file d'attente alias est résolue en une autre file d'attente, les vérifications ne sont pas effectuées sur la file d'attente sous-jacente. La vérification des droits d'accès est effectuée différemment pour les rubriques et les files d'attente.

Fermeture d'un abonnement

Un contrôle de sécurité supplémentaire est effectué si vous fermez un abonnement à l'aide de l'option MQCO_REMOVE_SUB si vous n'avez pas créé l'abonnement sous ce descripteur.

Un contrôle de sécurité est effectué pour vous assurer que vous disposez des droits appropriés pour effectuer cette opération, car l'action entraîne la suppression de l'abonnement. Si les attributs de sécurité associés au noeud de rubrique indiquent que l'utilisateur dispose de droits d'accès, l'accès est accordé. Si ce n'est pas le cas, le noeud parent de l'arborescence est pris en compte pour déterminer si l'utilisateur a le droit de fermer l'abonnement. La récursivité se poursuit jusqu'à ce que les droits soient accordés ou que le noeud racine soit atteint.

Définition, modification et suppression d'un abonnement

Aucun contrôle de sécurité d'abonnement n'est effectué lorsqu'un abonnement est créé de manière administrative au lieu d'utiliser une demande d'API MQSUB . Ce droit a déjà été accordé à l'administrateur via la commande.

Des contrôles de sécurité sont effectués pour s'assurer que les publications peuvent être placées dans la file d'attente de destination associée à l'abonnement. Les vérifications sont effectuées de la même manière que pour une demande MQSUB .

L'ID utilisateur utilisé pour ces contrôles de sécurité dépend de la commande émise. Si le paramètre **SUBUSER** est spécifié, il affecte la manière dont la vérification est effectuée, comme illustré dans la Tableau 81, à la page 488:

Tableau 81. ID utilisateur utilisés pour les contrôles de sécurité des commandes

Commande	SUBUSER indiqué et vide	SUBUSER indiqué et terminé	SUBUSER non indiqué
	Utiliser l'ID administrateur		Utiliser l'ID utilisateur de l'abonnement LIKE
	Utiliser l'ID administrateur		Utilisez l'ID.DEFAULT.SU utilisateurB de SYSTEMabonnement -si la zone est vide, utilisez l'ID administrateur
	Utiliser l'ID administrateur		Utiliser l'ID utilisateur de l'abonnement existant

Le seul contrôle de sécurité effectué lors de la suppression d'abonnements à l'aide de la commande DELETE SUB est le contrôle de sécurité de la commande.

Exemple de configuration de la sécurité de publication / abonnement

Cette section décrit un scénario dans lequel le contrôle d'accès est configuré sur les rubriques de manière à permettre l'application du contrôle de sécurité selon les besoins.

Accorder l'accès à un utilisateur pour s'abonner à une rubrique

Cette rubrique est la première d'une liste de tâches qui vous indique comment accorder l'accès aux rubriques à plusieurs utilisateurs.

Pourquoi et quand exécuter cette tâche

Cette tâche suppose qu'aucun objet de rubrique d'administration n'existe et qu'aucun profil n'a été défini pour l'abonnement ou la publication. Les applications créent de nouveaux abonnements, plutôt que de reprendre des abonnements existants, et utilisent uniquement la chaîne de rubrique.

Une application peut s'abonner en fournissant un objet de rubrique, une chaîne de rubrique ou une combinaison des deux. Quelle que soit la façon dont l'application sélectionne, l'effet est de créer un

abonnement à un certain point de l'arborescence de rubriques. Si ce point de l'arborescence de rubriques est représenté par un objet de rubrique d'administration, un profil de sécurité est vérifié en fonction du nom de cet objet de rubrique.

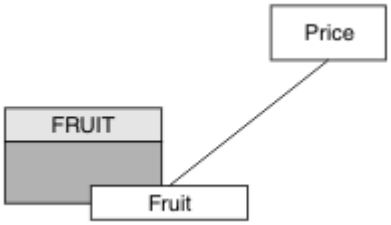


Figure 23. Exemple d'accès à un objet de rubrique

Tableau 82. Exemple d'accès à un objet de rubrique

Topic	Accès à l'abonnement requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun
Prix / Fruits	USER1	fruit

Définissez un nouvel objet de rubrique comme suit:

Procédure

1. Exécutez la commande MQSC `DEF TOPIC(FRUIT) TOPICSTR('Price/Fruit')`.
2. Accordez l'accès comme suit:

   - z/OS **z/OS** :

     Accordez l'accès à USER1 pour vous abonner à la rubrique `"Price/Fruit"` en accordant à l'utilisateur l'accès au profil `hlq.SUBSCRIBE.FRUIT` . Pour ce faire, utilisez les commandes RACF suivantes:

     ```
     RDEFINE MXTOPIC hlq.SUBSCRIBE.FRUIT UACC(NONE)
     PERMIT hlq.SUBSCRIBE.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
     ```

   - Autres plateformes:

     Accordez l'accès à USER1 pour vous abonner à la rubrique `"Price/Fruit"` en accordant à l'utilisateur l'accès à l'objet `FRUIT` . Pour ce faire, utilisez la commande d'autorisation pour la plateforme:

     ULW **Windows, systèmes UNIX and Linux**

     ```
     setmqaut -t topic -n FRUIT -p USER1 +sub
     ```

     IBM i **IBM i**

     ```
     GRTMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
     ```

Résultats

Lorsque USER1 tente de s'abonner à la rubrique `"Price/Fruit"` , le résultat est un succès.

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit"` , le résultat est un échec avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- z/OS Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- ULW Sur les autres plateformes, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

- IBM i Sous IBMi, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Notez qu'il s'agit d'une illustration de ce que vous voyez ; pas de tous les champs.

Accorder l'accès à un utilisateur pour s'abonner à une rubrique plus en profondeur dans l'arborescence

Cette rubrique est la deuxième d'une liste de tâches qui vous indique comment accorder l'accès aux rubriques à plusieurs utilisateurs.

Avant de commencer

Cette rubrique utilise la configuration décrite dans «Accorder l'accès à un utilisateur pour s'abonner à une rubrique», à la page 488.

Pourquoi et quand exécuter cette tâche

Si le point dans l'arborescence de rubriques où l'application effectue l'abonnement n'est pas représenté par un objet de rubrique d'administration, déplacez l'arborescence vers le haut jusqu'à ce que l'objet de rubrique d'administration parent le plus proche soit localisé. Le profil de sécurité est vérifié en fonction du nom de cet objet de rubrique.

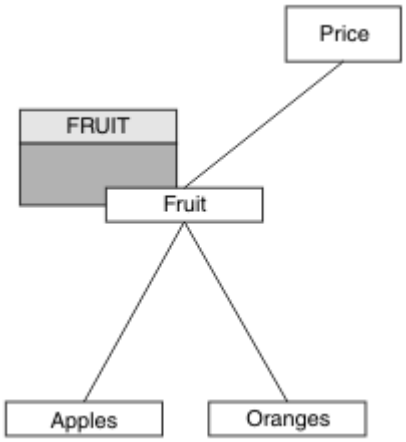


Figure 24. Exemple d'octroi d'accès à une rubrique dans une arborescence de rubriques

Tableau 83. Exigences d'accès pour des exemples de rubriques et d'objets de rubrique

Topic		Accès à l'abonnement requis	Objet de rubrique
Prix		Aucun utilisateur	Aucun
	Prix / Fruits	USER1	fruit
	Prix / Fruits / Pommes	USER1	
	Prix / Fruits / Oranges	USER1	

Dans la tâche précédente, USER1 a été autorisé à s'abonner à la rubrique `"Price/Fruit"` en lui accordant l'accès au profil `hlq.SUBSCRIBE.FRUIT` sur z/OS et l'accès au profil FRUIT sur d'autres plateformes. Ce profil unique accorde également à USER1 l'accès pour s'abonner à `"Price/Fruit/Apples"`, `"Price/Fruit/Oranges"` et `"Price/Fruit/#"`.

Lorsque USER1 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le résultat est un succès.

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le résultat est un échec avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- Sur les autres plateformes, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Apples"
```

Notez ce qui suit :

- Les messages que vous recevez sur z/OS sont identiques à ceux reçus lors de la tâche précédente car les mêmes objets de rubrique et les mêmes profils contrôlent l'accès.
- Le message d'événement que vous recevez sur d'autres plateformes est similaire à celui reçu lors de la tâche précédente, mais la chaîne de rubrique réelle est différente.

Accorder à un autre utilisateur l'accès permettant de s'abonner uniquement à la rubrique située plus en profondeur dans l'arborescence

Cette rubrique est la troisième d'une liste de tâches qui vous indique comment accorder l'accès à l'abonnement à des rubriques par plusieurs utilisateurs.

Avant de commencer

Cette rubrique utilise la configuration décrite dans «Accorder l'accès à un utilisateur pour s'abonner à une rubrique plus en profondeur dans l'arborescence», à la page 490.

Pourquoi et quand exécuter cette tâche

Dans la tâche précédente, l'accès à la rubrique `"Price/Fruit/Apples"`a été refusé USER2 . Cette rubrique vous explique comment accorder l'accès à cette rubrique, mais pas à d'autres rubriques.

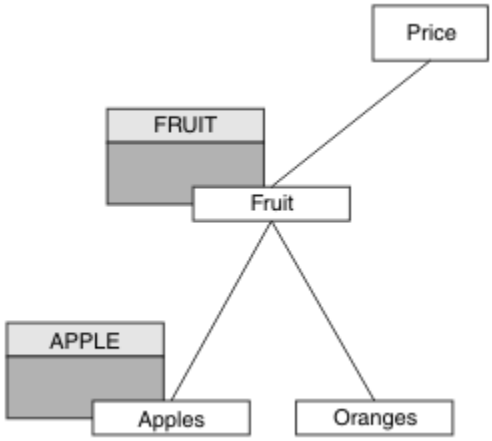


Figure 25. Octroi de l'accès à des rubriques spécifiques dans une arborescence de rubriques

Tableau 84. Exigences d'accès pour des exemples de rubriques et d'objets de rubrique

Topic	Accès à l'abonnement requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun
Prix / Fruits	USER1	fruit
Prix / Fruits / Pommes	USER1 et USER2	Apple
Prix / Fruits / Oranges	USER1	

Définissez un nouvel objet de rubrique comme suit:

Procédure

1. Exécutez la commande MQSC `DEF TOPIC(APPLE) TOPICSTR('Price/Fruit/Apples')`.
2. Accordez l'accès comme suit:

- z/OS **z/OS** :

  Dans la tâche précédente, USER1 a été autorisé à s'abonner à la rubrique `"Price/Fruit/Apples"` en accordant à l'utilisateur l'accès au profil `hlq.SUBSCRIBE.FRUIT` .

  Ce profil unique a également accordé à USER1 l'accès pour s'abonner à `"Price/Fruit/Oranges"` `"Price/Fruit/#"` et cet accès reste même avec l'ajout du nouvel objet de rubrique et des profils qui lui sont associés.

  Accordez l'accès à USER2 pour vous abonner à la rubrique `"Price/Fruit/Apples"` en accordant à l'utilisateur l'accès au profil `hlq.SUBSCRIBE.APPLE` . Pour ce faire, utilisez les commandes RACF suivantes:

  ```
  RDEFINE MXTOPIC hlq.SUBSCRIBE.APPLE UACC(NONE)
  PERMIT hlq.SUBSCRIBE.FRUIT APPLE(MXTOPIC) ID(USER2) ACCESS(ALTER)
  ```

- Autres plateformes:

  Dans la tâche précédente, USER1 a été autorisé à s'abonner à la rubrique `"Price/Fruit/Apples"` en accordant à l'utilisateur un accès d'abonnement au profil `FRUIT` .

  Ce profil unique a également accordé à USER1 l'accès pour s'abonner à `"Price/Fruit/Oranges"` et `"Price/Fruit/#"`, et cet accès reste même avec l'ajout du nouvel objet de rubrique et des profils qui lui sont associés.

  Accordez l'accès à USER2 pour vous abonner à la rubrique `"Price/Fruit/Apples"` en accordant à l'utilisateur l'accès par abonnement au profil `APPLE` . Pour ce faire, utilisez la commande d'autorisation pour la plateforme:

  ULW **Windows, systèmes UNIX and Linux**

  ```
  setmqaut -t topic -n APPLE -p USER2 +sub
  ```

  IBM i **IBM i**

  ```
  GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER2) AUT(*SUB)
  ```

Résultats

Sous z/OS, lorsque USER1 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le premier contrôle de sécurité du profil `hlq.SUBSCRIBE.APPLE` échoue, mais lorsqu'il remonte l'arborescence, le profil `hlq.SUBSCRIBE.FRUIT` permet à USER1 de s'abonner. L'abonnement aboutit donc et aucun code retour n'est envoyé à l'appel MQSUB. Toutefois, un message RACF `ICH` est généré pour la première vérification:

```
ICH408I USER(USER1   ) ...
  hlq.SUBSCRIBE.APPLE ...
```

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le résultat est un succès car le contrôle de sécurité réussit sur le premier profil.

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit/Oranges"` , le résultat est un échec avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- z/OS Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

  ```
  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.FRUIT ...
  ```

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- ULW Sur les plateformes Windows, UNIX and Linux , l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

- IBM i Sous IBMi, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

L'inconvénient de cette configuration est que, sous z/OS, vous recevez des messages ICH supplémentaires sur la console. Vous pouvez éviter cela si vous sécurisez l'arborescence de rubriques d'une manière différente.

Modifier le contrôle d'accès pour éviter les messages supplémentaires

Cette rubrique est la quatrième d'une liste de tâches qui vous indique comment accorder à plusieurs utilisateurs l'accès pour s'abonner à des rubriques et éviter des messages RACF ICH408I supplémentaires sur z/OS.

Avant de commencer

Cette rubrique améliore la configuration décrite dans «Accorder à un autre utilisateur l'accès permettant de s'abonner uniquement à la rubrique située plus en profondeur dans l'arborescence», à la page 492 afin d'éviter des messages d'erreur supplémentaires.

Pourquoi et quand exécuter cette tâche

Cette rubrique vous explique comment accorder l'accès à des rubriques plus en profondeur dans l'arborescence et comment supprimer l'accès à la rubrique située en bas de l'arborescence lorsqu'aucun utilisateur n'en a besoin.

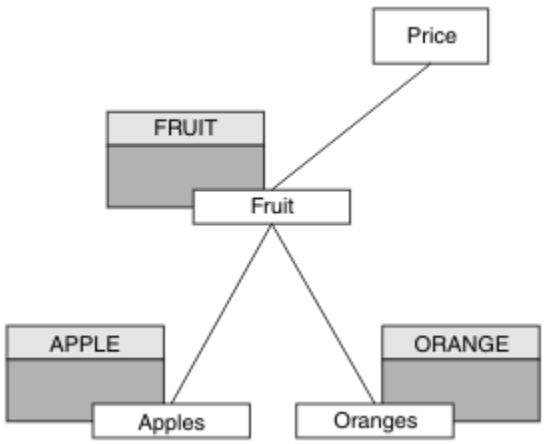


Figure 26. Exemple d'octroi de contrôle d'accès pour éviter des messages supplémentaires.

Définissez un nouvel objet de rubrique comme suit:

Procédure

1. Exécutez la commande MQSC `DEF TOPIC(ORANGE) TOPICSTR('Price/Fruit/Oranges')`.
2. Accordez l'accès comme suit:

   - z/OS **z/OS** :

     Définissez un nouveau profil et ajoutez l'accès à ce profil et aux profils existants. Pour ce faire, utilisez les commandes RACF suivantes:

     ```
     RDEFINE MXTOPIC hlq.SUBSCRIBE.ORANGE UACC(NONE)
     PERMIT hlq.SUBSCRIBE.ORANGE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
     PERMIT hlq.SUBSCRIBE.APPLE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
     ```

   - Autres plateformes:

     Configurez l'accès équivalent à l'aide des commandes d'autorisation pour la plateforme:

     ULW **Windows, systèmes UNIX and Linux**

     ```
     setmqaut -t topic -n ORANGE -p USER1 +sub
     setmqaut -t topic -n APPLE -p USER1 +sub
     ```

     IBM i **IBM i**

     ```
     GRTMQAUT OBJ(ORANGE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
     GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
     ```

Résultats

Sous z/OS, lorsque USER1 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le premier contrôle de sécurité sur le profil `hlq.SUBSCRIBE.APPLE` aboutit.

De même, lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit/Apples"` , le résultat est un succès car le contrôle de sécurité réussit sur le premier profil.

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Fruit/Oranges"` , le résultat est un échec avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- z/OS Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

  ```
  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.ORANGE ...

  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.FRUIT ...

  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
  ```

- ULW Sur les autres plateformes, l'événement d'autorisation suivant:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit/Oranges"
  ```

- IBM i Sous IBMi, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

Accorder l'accès à un utilisateur pour la publication dans une rubrique

Cette rubrique est la première d'une liste de tâches qui vous indique comment accorder l'accès à la publication de rubriques à plusieurs utilisateurs.

Pourquoi et quand exécuter cette tâche

Cette tâche suppose qu'aucun objet de rubrique d'administration n'existe à droite de l'arborescence de rubriques et qu'aucun profil n'a été défini pour la publication. L'hypothèse utilisée est que les diffuseurs utilisent uniquement la chaîne de rubrique.

Une application peut publier dans une rubrique en fournissant un objet de rubrique, une chaîne de rubrique ou une combinaison des deux. Quel que soit le mode de sélection de l'application, l'effet est la publication à un certain point de l'arborescence de rubriques. Si ce point de l'arborescence de rubriques est représenté par un objet de rubrique d'administration, un profil de sécurité est vérifié en fonction du nom de cet objet de rubrique. Exemple :

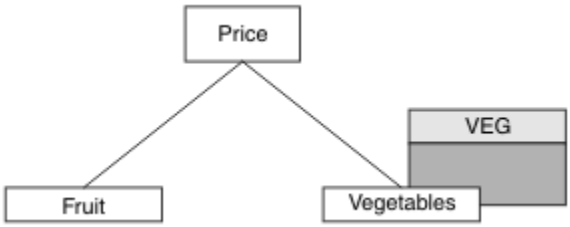


Figure 27. Octroi de l'accès en publication à une rubrique

Tableau 85. Exemple de conditions d'accès à la publication

Topic	Accès à la publication requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun
Prix / Légumes	USER1	VEG

Définissez un nouvel objet de rubrique comme suit:

Procédure

1. Exécutez la commande MQSC `DEF TOPIC(VEG) TOPICSTR('Price/Vegetables')`.
2. Accordez l'accès comme suit:
   - z/OS **z/OS** :

     Accordez l'accès à USER1 pour publier dans la rubrique `"Price/Vegetables"` en accordant à l'utilisateur l'accès au profil `hlq.PUBLISH.VEG` . Pour ce faire, utilisez les commandes RACF suivantes:

     ```
     RDEFINE MXTOPIC hlq.PUBLISH.VEG UACC(NONE)
     PERMIT hlq.PUBLISH.VEG CLASS(MXTOPIC) ID(USER1) ACCESS(UPDATE)
     ```

- Autres plateformes:

  Accordez l'accès à USER1 pour publier dans la rubrique "Price/Vegetables" en accordant à l'utilisateur l'accès au profil VEG . Pour ce faire, utilisez la commande d'autorisation pour la plateforme:

  ULW **Windows, systèmes UNIX and Linux**

  ```
  setmqaut -t topic -n VEG -p USER1 +pub
  ```

  IBM i **IBM i**

  ```
  GRTMQAUT OBJ(VEG) OBJTYPE(*TOPIC) USER(USER1) AUT(*PUB)
  ```

Résultats

Lorsque USER1 tente de publier dans la rubrique "Price/Vegetables" , le résultat est un succès, c'est-à-dire que l'appel MQOPEN aboutit.

Lorsque USER2 tente de publier dans la rubrique "Price/Vegetables" , l'appel MQOPEN échoue avec un message MQRC_NOT_AUTHORIZED et:

- z/OS Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

  ```
  ICH408I USER(USER2   ) ...
    hlq.PUBLISH.VEG ...

  ICH408I USER(USER2   ) ...
    hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
  ```

- ULW Sur les autres plateformes, l'événement d'autorisation suivant:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
  TopicString           "Price/Vegetables"
  ```

- IBM i Sous IBMi, l'événement d'autorisation suivant:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
  TopicString           "Price/Vegetables"
  ```

Notez qu'il s'agit d'une illustration de ce que vous voyez ; pas de tous les champs.

Accorder l'accès à un utilisateur pour publier dans une rubrique plus en profondeur dans l'arborescence

Cette rubrique est la deuxième d'une liste de tâches qui vous indique comment accorder l'accès à la publication à des rubriques par plusieurs utilisateurs.

Avant de commencer

Cette rubrique utilise la configuration décrite dans «Accorder l'accès à un utilisateur pour la publication dans une rubrique», à la page 496.

Pourquoi et quand exécuter cette tâche

Si le point de l'arborescence de rubriques où l'application publie n'est pas représenté par un objet de rubrique d'administration, déplacez l'arborescence vers le haut jusqu'à ce que l'objet de rubrique d'administration parent le plus proche soit localisé. Le profil de sécurité est vérifié en fonction du nom de cet objet de rubrique.

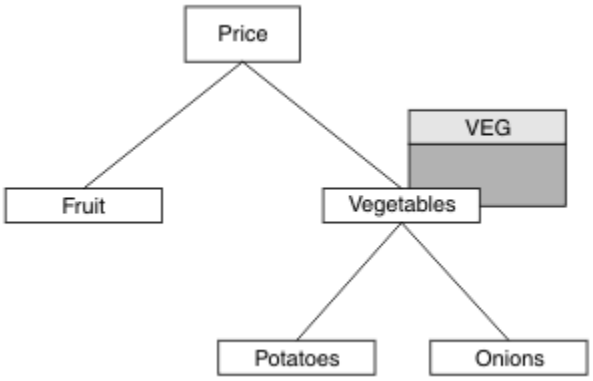


Figure 28. Octroi de l'accès en publication à une rubrique dans une arborescence de rubriques

Tableau 86. Exemple de conditions d'accès à la publication

Topic	Accès à l'abonnement requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun
Prix / Légumes	USER1	VEG
Prix / Légumes / Pommes de terre	USER1	
Prix / Légumes / Oignons	USER1	

Dans la tâche précédente, USER1 a été autorisé à publier la rubrique `"Price/Vegetables/Potatoes"` en lui accordant l'accès au profil `hlq.PUBLISH.VEG` sur z/OS ou l'accès en publication au profil VEG sur d'autres plateformes. Ce profil unique accorde également à USER1 l'accès à la publication sur `"Price/Vegetables/Onions"`.

Lorsque USER1 tente de publier dans la rubrique `"Price/Vegetables/Potatoes"` , le résultat est un succès, c'est-à-dire que l'appel MQOPEN aboutit.

Lorsque USER2 tente de s'abonner à la rubrique `"Price/Vegetables/Potatoes"` , le résultat est un échec, c'est-à-dire que l'appel MQOPEN échoue avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.VEG ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- Sur les autres plateformes, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
```

```
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables/Potatoes"
```

Notez ce qui suit :

- Les messages que vous recevez sur z/OS sont identiques à ceux reçus lors de la tâche précédente car les mêmes objets de rubrique et les mêmes profils contrôlent l'accès.
- Le message d'événement que vous recevez sur d'autres plateformes est similaire à celui reçu lors de la tâche précédente, mais la chaîne de rubrique réelle est différente.

Accorder l'accès pour la publication et l'abonnement

Cette rubrique est la dernière d'une liste de tâches qui vous indique comment accorder l'accès à la publication et l'abonnement à des rubriques par plusieurs utilisateurs.

Avant de commencer

Cette rubrique utilise la configuration décrite dans «Accorder l'accès à un utilisateur pour publier dans une rubrique plus en profondeur dans l'arborescence», à la page 497.

Pourquoi et quand exécuter cette tâche

Dans une tâche précédente, USER1 a été autorisé à s'abonner à la rubrique `"Price/Fruit"`. Cette rubrique vous indique comment accorder l'accès à cet utilisateur pour publier dans cette rubrique.

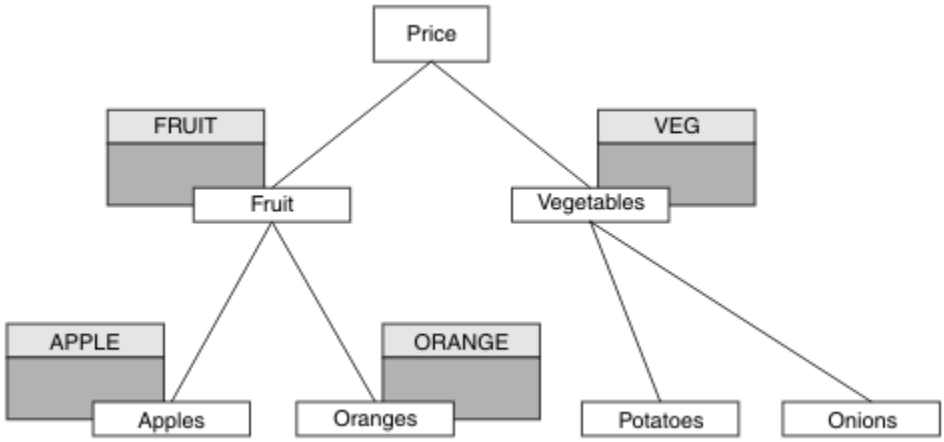


Figure 29. Octroi d'accès pour la publication et l'abonnement

Tableau 87. Exemple de conditions d'accès à la publication et à l'abonnement

Topic	Accès à l'abonnement requis	Accès à la publication requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun utilisateur	Aucun
Prix / Fruits	USER1	USER1	fruit
Prix / Fruits / Pommes	USER1 et USER2		Apple
Prix / Fruits / Oranges	USER1		ORANGE

Procédure

Accordez l'accès comme suit:

- z/OS **z/OS** :

  Dans une tâche antérieure, USER1 a été autorisé à s'abonner à la rubrique `"Price/Fruit"` en accordant à l'utilisateur l'accès au profil `hlq.SUBSCRIBE.FRUIT` .

  Pour publier dans la rubrique `"Price/Fruit"` , accordez l'accès à USER1 au profil `hlq.PUBLISH.FRUIT` . Pour ce faire, utilisez les commandes RACF suivantes:

```
RDEFINE MXTOPIC hlq.PUBLISH.FRUIT UACC(NONE)
PERMIT hlq.PUBLISH.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
```

- Autres plateformes:

  Accordez l'accès à USER1 pour publier dans la rubrique `"Price/Fruit"` en accordant à l'utilisateur l'accès en publication au profil `FRUIT` . Pour ce faire, utilisez la commande d'autorisation pour la plateforme:

  ULW **Windows, systèmes UNIX and Linux**

```
setmqaut -t topic -n FRUIT -p USER1 +pub
```

  IBM i **IBM i**

```
GRTMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*PUB)
```

Résultats

Sous z/OS, lorsque USER1 tente de publier dans la rubrique `"Price/Fruit"` , le contrôle de sécurité de l'appel MQOPEN passe.

Lorsque USER2 tente de publier à la rubrique `"Price/Fruit"` , le résultat est un échec avec un message `MQRC_NOT_AUTHORIZED` , ainsi que:

- z/OS Sous z/OS, les messages suivants affichés sur la console indiquent le chemin de sécurité complet via l'arborescence de rubriques qui a été tentée:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- ULW Sur les plateformes Windows, UNIXet Linux , l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

- IBM i Sous IBMi, l'événement d'autorisation suivant:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

En suivant l'ensemble complet de ces tâches, vous attribuez à USER1 et USER2 les droits d'accès suivants pour la publication et l'abonnement aux rubriques répertoriées:

Tableau 88. Liste complète des droits d'accès résultant d'exemples de sécurité

Topic	Accès à l'abonnement requis	Accès à la publication requis	Objet de rubrique
Prix	Aucun utilisateur	Aucun utilisateur	Aucun
Prix / Fruits	USER1	USER1	fruit
Prix / Fruits / Pommes	USER1 et USER2		Apple
Prix / Fruits / Oranges	USER1		ORANGE
Prix / Légumes		USER1	VEG
Prix / Légumes / Pommes de terre			
Prix / Légumes / Oignons			

Lorsque vous avez des exigences différentes en matière d'accès de sécurité à différents niveaux de l'arborescence de rubriques, une planification minutieuse garantit que vous ne recevez pas d'avertissements de sécurité superflus dans le journal de la console z/OS . La configuration de la sécurité au niveau approprié dans l'arborescence permet d'éviter les messages de sécurité trompeurs.

Sécurité des abonnements

MQSO_ALTERNATE_USER_AUTHORITY

La zone ID AlternateUsercontient un identificateur utilisateur à utiliser pour valider cet appel MQSUB. L'appel peut aboutir uniquement si cet ID AlternateUserest autorisé à s'abonner à la rubrique avec les options d'accès spécifiées, que l'ID utilisateur sous lequel l'application s'exécute soit autorisé ou non à le faire.

MQSO_SET_IDENTITY_CONTEXT

L'abonnement permet d'utiliser le jeton de comptabilité et les données d'identité d'application fournies dans les zones PubAccountingToken et PubApplIdentityData .

Si cette option est spécifiée, la même vérification d'autorisation est effectuée comme si la file d'attente de destination était accessible à l'aide d'un appel MQOPEN avec MQOO_SET_IDENTITY_CONTEXT, sauf dans le cas où l'option MQSO_MANAGED est également utilisée, auquel cas aucune vérification d'autorisation n'est effectuée sur la file d'attente de destination.

Si cette option n'est pas spécifiée, les informations de contexte par défaut sont associées aux publications envoyées à cet abonné comme suit:

Tableau 89. Informations de contexte de publication par défaut	
Zone dans MQMD	**Valeur utilisée**
UserIdentifier	ID utilisateur associé à l'abonnement (voir la zone SUBUSER sur DISPLAY SBSTATUS) au moment de la publication.
AccountingToken	Déterminé à partir de l'environnement si possible ; défini sur MQACT_NONE dans le cas contraire.
ApplIdentityData	Mettez à blanc.

Cette option est valide uniquement avec MQSO_CREATE et MQSO_ALTER. Si elles sont utilisées avec MQSO_RESUME, les zones PubAccountingToken et PubApplIdentityData sont ignorées, de sorte que cette option n'a aucun effet.

Si un abonnement est modifié sans utiliser cette option alors que l'abonnement avait précédemment fourni des informations de contexte d'identité, des informations de contexte par défaut sont générées pour l'abonnement modifié.

Si un abonnement permettant à différents ID utilisateur de l'utiliser avec l'option MQSO_ANY_USERID, est repris par un autre ID utilisateur, le contexte d'identité par défaut est généré pour le nouvel ID utilisateur propriétaire de l'abonnement et toutes les publications suivantes sont distribuées contenant le nouveau contexte d'identité.

AlternateSecurityId

Il s'agit d'un identificateur de sécurité qui est transmis avec l'ID AlternateUserau service d'autorisation pour permettre l'exécution des vérifications d'autorisation appropriées. L'ID AlternateSecurityest utilisé uniquement si MQSO_ALTERNATE_USER_AUTHORITY est spécifié et que la zone d'ID AlternateUsern'est pas entièrement vide jusqu'au premier caractère null ou jusqu'à la fin de la zone.

Option d'abonnement MQSO_ANY_USERID

Lorsque MQSO_ANY_USERID est spécifié, l'identité de l'abonné n'est pas limitée à un seul ID utilisateur. Cela permet à tout utilisateur de modifier ou de reprendre l'abonnement lorsqu'il dispose des droits appropriés. Un seul utilisateur peut disposer de l'abonnement à la fois. Une tentative de reprise de l'utilisation d'un abonnement actuellement utilisé par une autre application entraîne l'échec de l'appel avec MQRC_SUBSCRIPTION_IN_USE.

Pour ajouter cette option à un abonnement existant, l'appel MQSUB (à l'aide de MQSO_ALTER) doit provenir du même ID utilisateur que l'abonnement d'origine.

Si un appel MQSUB fait référence à un abonnement existant avec MQSO_ANY_USERID défini et que l'ID utilisateur est différent de l'abonnement d'origine, l'appel aboutit uniquement si le nouvel ID utilisateur est autorisé à s'abonner à la rubrique. Une fois l'opération terminée, les futures publications destinées à cet abonné sont placées dans la file d'attente de l'abonné avec le nouvel ID utilisateur défini dans la publication.

ID_UTILISATEUR_FIXE_MQSO_FIXE

Lorsque MQSO_FIXED_USERID est spécifié, l'abonnement ne peut être modifié ou repris que par un seul ID utilisateur propriétaire. Cet ID utilisateur est le dernier ID utilisateur à modifier l'abonnement qui définit cette option, supprimant ainsi l'option MQSO_ANY_USERID, ou si aucune modification n'a eu lieu, il s'agit de l'ID utilisateur qui a créé l'abonnement.

Si une instruction MQSUB fait référence à un abonnement existant avec MQSO_ANY_USERID défini et modifie l'abonnement (à l'aide de MQSO_ALTER) pour utiliser l'option MQSO_FIXED_USERID, l'ID utilisateur de l'abonnement est désormais corrigé au niveau de ce nouvel ID utilisateur. L'appel aboutit uniquement si le nouvel ID utilisateur est autorisé à s'abonner à la rubrique.

Si un ID utilisateur autre que celui enregistré comme propriétaire d'un abonnement tente de reprendre ou de modifier un abonnement MQSO_FIXED_USERID, l'appel échoue avec MQRC_IDENTITY_MISMATCH. L'ID utilisateur propriétaire d'un abonnement peut être affiché à l'aide de la commande DISPLAY SBSTATUS.

Si ni MQSO_ANY_USERID ni MQSO_FIXED_USERID n'est spécifié, la valeur par défaut est MQSO_FIXED_USERID.

Sécurité de publication / abonnement entre les gestionnaires de files d'attente

Les messages internes de publication / abonnement, tels que les abonnements de proxy et les publications, sont placés dans des files d'attente système de publication / abonnement à l'aide de règles de sécurité de canal normales. Les informations et les diagrammes de cette rubrique mettent en évidence les différents processus et ID utilisateur impliqués dans la distribution de ces messages.

Contrôle d'accès local

L'accès aux rubriques pour la publication et les abonnements est régi par des définitions et des règles de sécurité locales décrites dans Sécurité de publication / abonnement. Sous z/OS, aucun objet de rubrique local n'est requis pour établir le contrôle d'accès. Aucune rubrique locale n'est requise pour le contrôle d'accès sur d'autres plateformes. Les administrateurs peuvent choisir d'appliquer le contrôle d'accès aux objets de rubrique en cluster, qu'ils existent ou non dans le cluster.

Les administrateurs système sont responsables du contrôle d'accès sur leur système local. Ils doivent faire confiance aux administrateurs des autres membres de la hiérarchie ou des collectivités de cluster pour qu'ils soient responsables de leur stratégie de contrôle d'accès. Étant donné que le contrôle d'accès est défini pour chaque machine distincte, il risque d'être fastidieux si un contrôle de niveau fin est nécessaire. Il peut ne pas être nécessaire d'imposer un contrôle d'accès, ou le contrôle d'accès peut être défini sur des objets de haut niveau dans l'arborescence de rubriques. Un contrôle d'accès de niveau fin peut être défini pour chaque subdivision de l'espace de nom de sujet.

Création d'un abonnement de proxy

La confiance d'une organisation pour la connexion de son gestionnaire de files d'attente à votre gestionnaire de files d'attente est confirmée par des moyens d'authentification de canal normaux. Si cette organisation digne de confiance est également autorisée à effectuer une publication / abonnement distribué, une vérification des droits d'accès est effectuée. La vérification est effectuée lorsque le canal insère un message dans une file d'attente de publication / abonnement distribuée. Par exemple, si un message est inséré dans la file d'attente `SYSTEM.INTER.QMGR.CONTROL` . L'ID utilisateur pour la vérification des droits d'accès à la file d'attente dépend des valeurs PUTAUT du canal récepteur. Par exemple, l'ID utilisateur du canal, `MCAUSER`, le contexte de message, en fonction de la valeur et de la plateforme. Pour plus d'informations sur la sécurité des canaux, voir Sécurité des canaux.

Les abonnements de proxy sont effectués avec l'ID utilisateur de l'agent de publication / abonnement réparti sur le gestionnaire de files d'attente éloignées. Par exemple, `QM2` dans Figure 30, à la page 504. L'utilisateur est alors facilement autorisé à accéder aux profils d'objet de rubrique locaux, car cet ID utilisateur est défini dans le système et il n'y a donc pas de conflit de domaine.

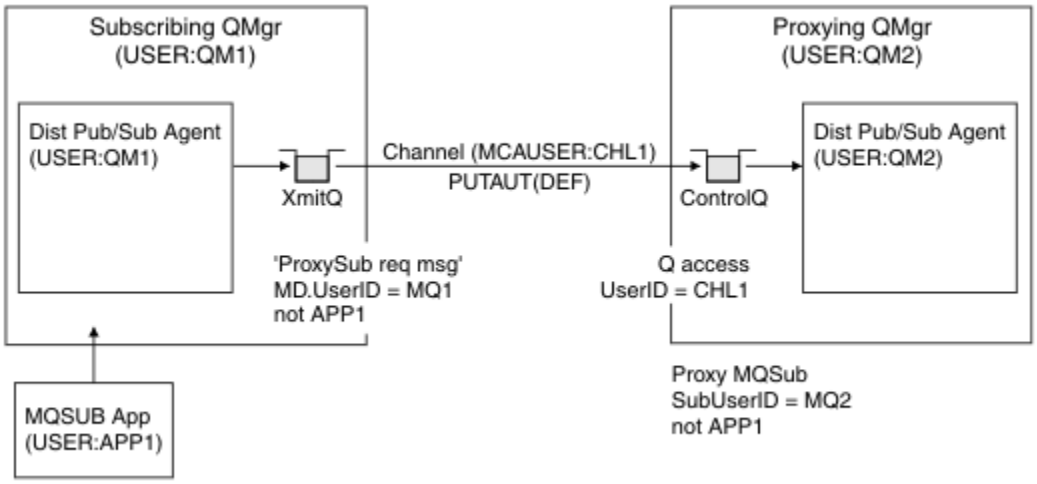


Figure 30. sécurité de l'abonnement de proxy, création d'un abonnement

Renvoi de publications distantes

Lorsqu'une publication est créée sur le gestionnaire de files d'attente de publication, une copie de la publication est créée pour tout abonnement de proxy. Le contexte de la publication copiée contient le contexte de l'ID utilisateur qui a effectué l'abonnement ; QM2 dans Figure 31, à la page 504. L'abonnement proxy est créé avec une file d'attente de destination qui est une file d'attente éloignée, de sorte que le message de publication est résolu dans une file d'attente de transmission.

La confiance d'une organisation pour la connexion de son gestionnaire de files d'attente, QM2, à un autre gestionnaire de files d'attente, QM1, est confirmée par des moyens d'authentification de canal normaux. Si cette organisation digne de confiance est alors autorisée à effectuer une publication / abonnement distribué, une vérification des droits d'accès est effectuée lorsque le canal place le message de publication dans la file d'attente de publication / abonnement distribué SYSTEM.INTER.QMGR.PUBS. L'ID utilisateur pour la vérification des droits d'accès à la file d'attente dépend de la valeur PUTAUT du canal récepteur (par exemple, l'ID utilisateur du canal, MCAUSER, le contexte de message, etc., en fonction de la valeur et de la plateforme). Pour plus d'informations sur la sécurité des canaux, voir Sécurité des canaux.

Lorsque le message de publication atteint le gestionnaire de files d'attente d'abonnement, une autre opération MQPUT sur la rubrique est effectuée sous l'autorité de ce gestionnaire de files d'attente et le contexte contenant le message est remplacé par le contexte de chacun des abonnés locaux tels qu'ils reçoivent chacun le message.

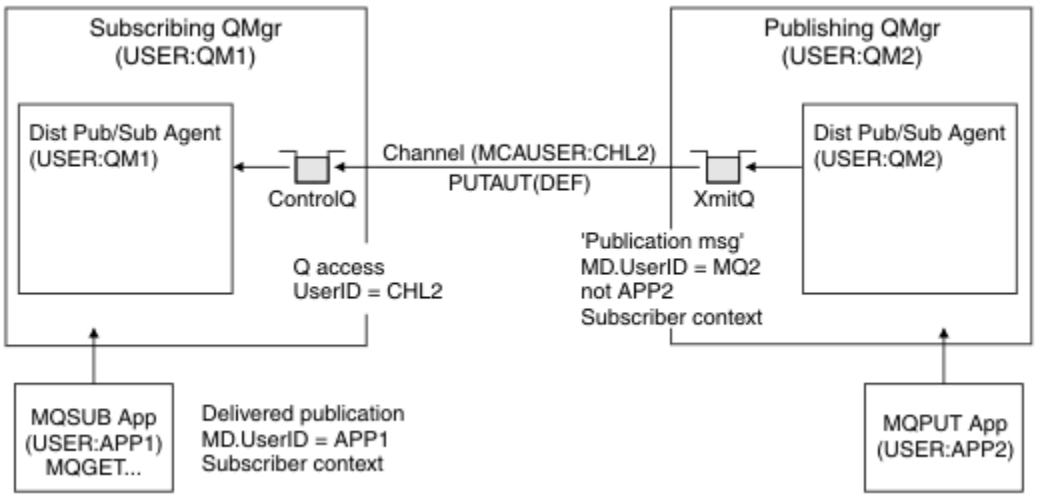


Figure 31. Sécurité des abonnements de proxy, transfert de publications

Sur un système où la sécurité est peu prise en compte, les processus de publication / abonnement distribué sont susceptibles de s'exécuter sous un ID utilisateur dans le groupe `mqm` , le paramètre `MCAUSER` sur un canal est vide (valeur par défaut) et les messages sont distribués aux différentes files d'attente système selon les besoins. Le système non sécurisé facilite la mise en place d'une preuve de concept pour illustrer la publication / l'abonnement distribué.

Sur un système où la sécurité est plus sérieusement prise en compte, ces messages internes sont soumis aux mêmes contrôles de sécurité que tout message passant par le canal.

Si le canal est configuré avec une valeur `MCAUSER` non vide et une valeur `PUTAUT` indiquant que `MCAUSER` doit être vérifié, l'accès aux files d'attente `SYSTEM.INTER.QMGR.*` doit être accordé à `MCAUSER` en question. S'il existe plusieurs gestionnaires de files d'attente éloignées, avec des canaux s'exécutant sous des ID `MCAUSER` différents, tous ces ID utilisateur doivent être autorisés à accéder aux files d'attente `SYSTEM.INTER.QMGR.*` . Des canaux s'exécutant sous des ID `MCAUSER` différents peuvent se produire, par exemple, lorsque plusieurs connexions hiérarchiques sont configurées sur un seul gestionnaire de files d'attente.

Si le canal est configuré avec une valeur `PUTAUT` spécifiant que le contexte du message est utilisé, l'accès aux files d'attente `SYSTEM.INTER.QMGR.*` est vérifié en fonction de l'ID utilisateur dans le message interne. Etant donné que tous ces messages sont insérés avec l'ID utilisateur de l'agent de publication / abonnement distribué à partir du gestionnaire de files d'attente qui envoie le message interne ou le message de publication (voir Figure 31, à la page 504), il ne s'agit pas d'un ensemble trop important d'ID utilisateur pour accorder l'accès aux différentes files d'attente système (une par gestionnaire de files d'attente éloignées), si vous souhaitez configurer votre sécurité de publication / abonnement distribué de cette manière. Il a toujours les mêmes problèmes que la sécurité de contexte de canal ; celui des différents domaines d'ID utilisateur et le fait que l'ID utilisateur dans le message peut ne pas être défini sur le système récepteur. Cependant, il s'agit d'un moyen tout à fait acceptable de fonctionner si nécessaire.

z/OS La sécurité des files d'attente système fournit la liste des files d'attente et l'accès requis pour configurer de manière sécurisée votre environnement de publication / abonnement distribué. Si des messages internes ou des publications ne parviennent pas à être insérés en raison de violations de sécurité, le canal écrit un message dans le journal de manière normale et les messages peuvent être envoyés à la file d'attente des messages non livrés en fonction du traitement normal des erreurs du canal.

Toute la messagerie inter-gestionnaire de files d'attente pour les besoins de la publication / abonnement distribué s'exécute à l'aide de la sécurité de canal normale.

Pour plus d'informations sur la restriction des publications et des abonnements de proxy au niveau des rubriques, voir Sécurité de publication / abonnement.

Utilisation des ID utilisateur par défaut avec une hiérarchie de gestionnaires de files d'attente

Si vous disposez d'une hiérarchie de gestionnaires de files d'attente s'exécutant sur des plateformes différentes et que vous utilisez des ID utilisateur par défaut, notez que ces ID utilisateur par défaut diffèrent d'une plateforme à l'autre et peuvent ne pas être connus sur la plateforme cible. Par conséquent, un gestionnaire de files d'attente s'exécutant sur une plateforme rejette les messages reçus des gestionnaires de files d'attente sur d'autres plateformes avec le code anomalie `MQRC_NOT_AUTHORIZED`.

Pour éviter que des messages ne soient rejetés, au minimum, les droits suivants doivent être ajoutés aux ID utilisateur par défaut utilisés sur d'autres plateformes:

- Droit *PUT *GET sur le système SYSTEM.BROKER. queues
- *PUB *SUB droit sur SYSTEM.BROKER. rubriques
- Droit *ADMCRT *ADMDLT *ADMCHG sur le système SYSTEM.BROKER.CONTROL.QUEUE .

Les ID utilisateur par défaut avec une hiérarchie de gestionnaires de files d'attente sont les suivants:

Plateforme	ID utilisateur par défaut
Windows	MUSR_MQADMIN
Systèmes UNIX and Linux	mqm
IBM i	QMQM
z/OS	ID utilisateur de l'espace adresse de l'initiateur de canal

Créez et accordez l'accès à l'ID utilisateur'qmqm's'il est associé de manière hiérarchique à un gestionnaire de files d'attente sur les plateformes IBM i for Queue Managers on Windows, UNIX, Linuxet z/OS .

Créez et accordez l'accès à l'ID utilisateur'mqm's'il est associé de manière hiérarchique à un gestionnaire de files d'attente sur les plateformes Windows, UNIXou Linux pour les gestionnaires de files d'attente sur les plateformes IBM i et z/OS .

Créez et accordez à l'utilisateur l'accès à l'ID utilisateur de l'espace adresse de l'initiateur de canal z/OS s'il est connecté de manière hiérarchique à un gestionnaire de files d'attente sur les plateformes z/OS for Queue Managers on Windows, UNIX, Linuxet IBM i .

Les ID utilisateur peuvent être sensibles à la casse. Le gestionnaire de files d'attente d'origine (si les systèmes IBM i, Windows, UNIXou Linux) force l'ID utilisateur à être en majuscules. Le gestionnaire de files d'attente de réception (si les systèmes Windows, UNIX ou Linux) force l'ID utilisateur à être en minuscules. Par conséquent, tous les ID utilisateur créés sur les systèmes UNIX and Linux doivent être créés en minuscules. Si un exit de message a été installé, la mise en majuscules ou en minuscules de l'ID utilisateur n'est pas forcée. Prenez soin de comprendre comment l'exit de message traite l'ID utilisateur.

Pour éviter les problèmes potentiels liés à la conversion des ID utilisateur:

- Sur les systèmes UNIX, Linux, and Windows , vérifiez que les ID utilisateur sont spécifiés en minuscules.
- Sous IBM i et z/OS, vérifiez que les ID utilisateur sont indiqués en majuscules.

V 9.1.0 Sécurité IBM MQ Console et REST API

La sécurité pour IBM MQ Console et REST API est configurée en éditant la configuration du serveur mqweb dans le fichier `mqwebuser.xml` .

Pourquoi et quand exécuter cette tâche

Vous pouvez suivre les actions utilisateur et auditer l'utilisation de IBM MQ Console et de REST API en examinant les fichiers journaux du serveur mqweb.

Les utilisateurs de IBM MQ Console et de REST API peuvent être authentifiés à l'aide des éléments suivants:

- Registre de base
- Registre LDAP
- Registre du système d'exploitation local
- SAF sous z/OS
- Tout autre type de registre pris en charge par WebSphere Liberty

Des rôles peuvent être affectés à des utilisateurs IBM MQ Console et à des utilisateurs REST API pour déterminer le niveau d'accès qui leur est accordé aux objets IBM MQ . Par exemple, pour effectuer des opérations de messagerie, le rôle `MQWebUser` doit être affecté aux utilisateurs. Pour plus d'informations sur les rôles disponibles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

Une fois qu'un rôle a été affecté à un utilisateur, un certain nombre de méthodes peuvent être utilisées pour l'authentification de l'utilisateur. Avec IBM MQ Console, les utilisateurs peuvent se connecter avec un nom d'utilisateur et un mot de passe ou utiliser l'authentification par certificat client. Avec REST API,

les utilisateurs peuvent utiliser l'authentification HTTP de base, l'authentification basée sur un jeton ou l'authentification par certificat client.

Procédure

1. Définissez le registre d'utilisateurs pour authentifier les utilisateurs et affectez à chaque utilisateur ou groupe un rôle pour autoriser les utilisateurs et les groupes à utiliser IBM MQ Console ou REST API. Pour plus d'informations, voir «Configuration des utilisateurs et des rôles», à la page 508
2. Choisissez comment les utilisateurs de IBM MQ Console s'authentifient auprès du serveur mqweb. Il n'est pas nécessaire d'utiliser la même méthode pour tous les utilisateurs:
   - Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'heure d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.
   - Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.
3. Choisissez comment les utilisateurs de REST API s'authentifient auprès du serveur mqweb. Il n'est pas nécessaire d'utiliser la même méthode pour tous les utilisateurs:
   - Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
   - Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524.

     Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Toutefois, si vous avez activé les connexions HTTP, vous pouvez autoriser l'utilisation d'un jeton LTPA émis pour une connexion HTTPS pour une connexion HTTP. Pour plus d'informations, voir Configuration du jeton LTPA.
   - Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.
4. Facultatif : Configurez le partage de ressources d'origine croisée pour REST API.

   Par défaut, un navigateur Web n'autorise pas les scripts, tels que JavaScript, à appeler REST API lorsque le script n'est pas de la même origine que le REST API. C'est-à-dire que les demandes d'origine croisée ne sont pas activées. Vous pouvez configurer le partage de ressources d'origine croisée (CORS) pour autoriser les demandes d'origine croisée à partir d'URL spécifiées. Pour plus d'informations, voir «Configuration de CORS pour REST API», à la page 527.
5. Facultatif : Configurez la validation de l'en-tête d'hôte pour IBM MQ Console et REST API.

   Vous pouvez configurer la validation de l'en-tête d'hôte et créer une liste autorisée de noms d'hôte et de ports pour vous assurer que seules les demandes contenant des en-têtes d'hôte spécifiques sont traitées par IBM MQ Console et REST API. Pour plus d'informations, voir «Configuration de la validation de l'en-tête d'hôte pour IBM MQ Console et REST API», à la page 528.

V 9.1.0 Configuration des utilisateurs et des rôles

Pour utiliser IBM MQ Console ou REST API, les utilisateurs doivent s'authentifier auprès d'un registre d'utilisateurs défini sur le serveur mqweb.

Pourquoi et quand exécuter cette tâche

Les utilisateurs authentifiés doivent être membres de l'un des groupes qui autorise l'accès aux fonctions de IBM MQ Console et REST API. Par défaut, le registre d'utilisateurs ne contient aucun utilisateur ; vous devez les ajouter en éditant le fichier `mqwebuser.xml` .

Lorsque vous configurez des utilisateurs et des groupes, vous devez d'abord configurer un registre d'utilisateurs pour authentifier les utilisateurs et les groupes. Ce registre d'utilisateurs est partagé entre IBM MQ Console et REST API. Vous pouvez contrôler si les utilisateurs et les groupes ont accès à IBM MQ Consoleet/ou à REST APIlorsque vous configurez des rôles pour vos utilisateurs et groupes.

Après avoir configuré le registre d'utilisateurs, vous configurez des rôles pour les utilisateurs et les groupes afin de leur accorder des autorisations. Plusieurs rôles sont disponibles, notamment des rôles spécifiques à l'utilisation de REST API for Managed File Transfer. Chaque rôle accorde un niveau d'accès différent. Pour plus d'informations, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

Un certain nombre d'exemples de fichiers XML sont fournis avec le serveur mqweb pour simplifier la configuration des utilisateurs et des groupes. Les utilisateurs qui connaissent bien la configuration de la sécurité dans WebSphere Liberty (WLP) peuvent préférer ne pas utiliser les exemples. WLP fournit d'autres fonctions d'autorisation en plus de celles décrites ici.

Procédure

- Configurez les utilisateurs et les groupes avec un registre de base à l'aide du fichier `basic_registry.xml` .

  Les noms d'utilisateur et les mots de passe du registre sont utilisés pour authentifier et autoriser les utilisateurs de IBM MQ Console et de REST API.

  Pour configurer un registre de base à l'aide de l'exemple de fichier `basic_registry.xml` , voir «Configuration d'un registre de base pour IBM MQ Console et REST API», à la page 509.
- Configurez les utilisateurs et les groupes avec un registre LDAP à l'aide du fichier `ldap_registry.xml` .

  Les noms d'utilisateur et les mots de passe du registre LDAP sont utilisés pour authentifier et autoriser l'utilisation de IBM MQ Console et de REST API.

  Pour configurer un registre LDAP à l'aide de l'exemple de fichier `ldap_registry.xml` , voir «Configuration d'un registre LDAP pour IBM MQ Console et REST API», à la page 513.
- ULW

  Configurez les utilisateurs et les groupes avec un registre de système d'exploitation local à l'aide du fichier `local_os_registry.xml` .

  Les noms d'utilisateur et les mots de passe du registre du système d'exploitation sont utilisés pour authentifier et autoriser les utilisateurs du IBM MQ Console et du REST API.

  Pour configurer un registre de système d'exploitation local à l'aide de l'exemple de fichier `local_os_registry.xml` , voir «Configuration d'un registre de système d'exploitation local pour IBM MQ Console et REST API», à la page 512.
- z/OS

  Configurez les utilisateurs et les groupes avec l'interface SAF (System Authorization Facility) sous z/OS à l'aide du fichier `zos_saf_registry.xml` .

  Les profils RACF, ou tout autre produit de sécurité, sont utilisés pour accorder aux utilisateurs et aux groupes l'accès aux rôles. Les noms d'utilisateur et les mots de passe de la base de données RACF sont utilisés pour authentifier et autoriser les utilisateurs de IBM MQ Console et REST API.

Pour configurer l'interface SAF à l'aide de l'exemple de fichier `zos_saf_registry.xml` , voir «Configuration d'un registre SAF pour IBM MQ Console et REST API», à la page 515.

- Désactivez la sécurité, y compris la possibilité d'accéder au IBM MQ Consoleou au REST API, à l'aide de HTTPS, à l'aide du fichier `no_security.xml` .

Que faire ensuite

Choisissez comment les utilisateurs s'authentifient:

IBM MQ Consoleoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'intervalle d'expiration pour le jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

REST APIoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524. Vous pouvez configurer l'intervalle d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

V 9.1.0 Configuration d'un registre de base pour IBM MQ Console et REST API

Vous pouvez configurer un registre de base dans le fichier `mqwebuser.xml` . Les noms d'utilisateur, les mots de passe et les rôles du fichier xml sont utilisés pour authentifier et autoriser les utilisateurs de IBM MQ Console et de REST API.

Avant de commencer

- Lorsque vous configurez des utilisateurs dans le registre de base, vous devez affecter un rôle à chaque utilisateur. Chaque rôle fournit différents niveaux de privilèges pour accéder à IBM MQ Console et REST API, et détermine le contexte de sécurité qui est utilisé lorsqu'une opération autorisée est tentée. Vous devez comprendre ces rôles avant de configurer le registre de base. Pour plus d'informations sur chacun des rôles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.
- Pour effectuer cette tâche, vous devez être un utilisateur disposant de privilèges suffisants pour éditer le fichier `mqwebuser.xml` :

– z/OS Sous z/OS, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` .

– Multi Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.

Procédure

1. Copiez l'exemple de fichier XML `basic_registry.xml` à partir de l'un des chemins suivants:

   - ULW Sous UNIX, Linux, and Windows: *MQ_INSTALLATION_PATH* `/web/mq/samp/configuration`
   - z/OS Sous z/OS: `PathPrefix /web/mq/samp/configuration`

   où `PathPrefix` est le chemin d'installation de IBM MQ Unix System Services Components.
2. Placez l'exemple de fichier dans le répertoire approprié:

   - ULW

     Sous UNIX, Linux, and Windows : *MQ_DATA_PATH*`/web/installations/`*installationName*`/servers/mqweb`
   - z/OS

     Sous z/OS : *WLP_user_directory*`/servers/mqweb`

     où *WLP_user_directory* est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb** pour créer la définition de serveur mqweb.
3. Facultatif : Si vous avez modifié des paramètres de configuration dans `mqwebuser.xml`, copiez-les dans l'exemple de fichier.
4. Supprimez le fichier `mqwebuser.xml` existant et renommez l'exemple de fichier en `mqwebuser.xml`.
5. Editez le nouveau fichier `mqwebuser.xml` pour ajouter des utilisateurs et des groupes dans les balises **basicRegistry** .

   N'oubliez pas que tout utilisateur disposant du rôle `MQWebUser` peut effectuer uniquement les opérations que l'ID utilisateur est autorisé à effectuer sur le gestionnaire de files d'attente. Par conséquent, l'ID utilisateur défini dans le registre doit avoir un ID utilisateur identique sur le système sur lequel IBM MQ est installé. Ces ID utilisateur doivent être dans le même cas, sinon le mappage entre les ID utilisateur peut échouer.

   Pour plus d'informations sur la configuration des registres d'utilisateurs de base, voir Configuration d'un registre d'utilisateurs de base pour Liberty dans la documentation WebSphere Liberty .
6. Affectez des rôles aux utilisateurs et aux groupes en éditant le fichier `mqwebuser.xml` :

   Plusieurs rôles sont disponibles pour autoriser les utilisateurs et les groupes à utiliser le IBM MQ Consoleet le REST API. Chaque rôle accorde un niveau d'accès différent. Pour plus d'informations, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

   - Pour affecter des rôles et accorder l'accès au IBM MQ Console, ajoutez vos utilisateurs et groupes entre les balises **security-role** appropriées dans les balises **<enterpriseApplication id="com.ibm.mq.console">** .
   - Pour affecter des rôles et accorder l'accès au REST API, ajoutez vos utilisateurs et groupes entre les balises **security-role** appropriées dans les balises **<enterpriseApplication id="com.ibm.mq.rest">** .

   Pour obtenir de l'aide sur le format des informations d'utilisateur et de groupe dans les balises **security-role** , voir les exemples.
7. Si vous avez fourni des mots de passe pour les utilisateurs dans `mqwebuser.xml`, vous devez les coder afin de les sécuriser à l'aide de la commande **securityUtility encoding** fournie par WebSphere Liberty. Pour plus d'informations, voir Liberty: commandesecurityUtility dans la documentation du produit WebSphere Liberty .

Exemple

Dans l'exemple suivant, le groupe MQWebAdminGroup est autorisé à accéder au IBM MQ Console avec le rôle MQWebAdmin. L'accès est accordé à l'utilisateur readeravec le rôle MQWebAdminROet à l'utilisateur guest avec le rôle MQWebUser:

```
<enterpriseApplication id="com.ibm.mq.console">
        <application-bnd>
            <security-role name="MQWebAdmin">
                <group name="MQWebAdminGroup" realm="defaultRealm"/>
            </security-role>
            <security-role name="MQWebAdminRO">
                <user name="reader" realm="defaultRealm"/>
            </security-role>
            <security-role name="MQWebUser">
                <user name="guest" realm="defaultRealm"/>
            </security-role>
        </application-bnd>
    </enterpriseApplication>
```

Dans l'exemple suivant, les utilisateurs reader et guest sont autorisés à accéder à IBM MQ Console. L'utilisateur user a accès à REST APIet tous les utilisateurs du groupe MQAdmin ont accès à IBM MQ Console et à REST API. L'utilisateur mftadmin est autorisé à accéder à REST API for MFT :

```
<enterpriseApplication id="com.ibm.mq.console">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="MQAdmin" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebAdminRO">
            <user name="reader" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebUser">
            <user name="guest" realm="defaultRealm"/>
        </security-role>
    </application-bnd>
</enterpriseApplication>

<enterpriseApplication id="com.ibm.mq.rest">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="MQAdmin" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebUser">
            <user name="user" realm="defaultRealm"/>
        </security-role>
        <security-role name="MFTWebAdmin">
            <user name="mftadmin" realm="defaultRealm"/>
        </security-role>
     </application-bnd>
</enterpriseApplication>
```

Que faire ensuite

Choisissez comment les utilisateurs s'authentifient:

IBM MQ Consoleoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'intervalle d'expiration pour le jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

REST APIoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524. Vous pouvez configurer l'intervalle d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

ULW V 9.1.0 Configuration d'un registre de système d'exploitation local pour IBM MQ Console et REST API

Vous pouvez configurer un registre de système d'exploitation local dans le fichier `mqwebuser.xml` . Les noms d'utilisateur et les mots de passe du système d'exploitation local sont utilisés pour authentifier et autoriser les utilisateurs du IBM MQ Console et du REST API.

Avant de commencer

- Pour l'authentification par certificat client avec la fonction d'authentification du système d'exploitation local, l'identité de l'utilisateur est le nom usuel (CN) à partir du nom distinctif (DN) du certificat client. Si l'identité de l'utilisateur n'existe pas en tant qu'utilisateur du système d'exploitation, la connexion par certificat client échoue et l'authentification par mot de passe est réactivée.
- Pour effectuer cette tâche, vous devez être un utilisateur privilégié.

Pourquoi et quand exécuter cette tâche

Avec un registre de système d'exploitation local, les utilisateurs et les groupes reçoivent automatiquement un rôle:

- Tout utilisateur faisant partie du groupe'mqm'ou du groupe'QMQMADM'sous IBM ise voit attribuer les rôles MQWebAdmin et MFTWebAdmin .
- Tous les autres utilisateurs reçoivent le rôle MQWebUser .

Pour plus d'informations sur ces rôles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

Un registre de système d'exploitation local ne peut être utilisé que sur UNIX, Linux, and Windows. Une fonction équivalente est fournie sur z/OS en configurant un registre SAF. Pour plus d'informations, voir «Configuration d'un registre SAF pour IBM MQ Console et REST API», à la page 515.

Procédure

1. Copiez l'exemple de fichier XML `local_os_registry.xml` à partir du chemin suivant:
   MQ_INSTALLATION_PATH`/web/mq/samp/configuration`
2. Placez l'exemple de fichier dans le répertoire suivant:
   MQ_DATA_PATH`/web/installations/`*installationName*`/servers/mqweb`
3. Facultatif : Si vous avez modifié des paramètres de configuration dans `mqwebuser.xml`, copiez-les dans l'exemple de fichier.

4. Supprimez le fichier `mqwebuser.xml` existant et renommez l'exemple de fichier en `mqwebuser.xml`.

Que faire ensuite

Choisissez comment les utilisateurs s'authentifient:

IBM MQ Consoleoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'intervalle d'expiration pour le jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

REST APIoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524. Vous pouvez configurer l'intervalle d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

V 9.1.0 Configuration d'un registre LDAP pour IBM MQ Console et REST API

Vous pouvez configurer un registre LDAP dans le fichier `mqwebuser.xml` . Les noms d'utilisateur et les mots de passe du registre LDAP sont utilisés pour authentifier et autoriser les utilisateurs de IBM MQ Console et de REST API.

Avant de commencer

- Lorsque vous configurez un registre LDAP, vous devez affecter un rôle à chaque utilisateur. Chaque rôle fournit différents niveaux de privilèges pour accéder à IBM MQ Console et REST API, et détermine le contexte de sécurité qui est utilisé lorsqu'une opération autorisée est tentée. Vous devez comprendre ces rôles avant de configurer le registre. Pour plus d'informations sur chacun des rôles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

  N'oubliez pas que tout utilisateur disposant du rôle `MQWebUser` peut effectuer uniquement les opérations que l'ID utilisateur est autorisé à effectuer sur le gestionnaire de files d'attente. Par conséquent, l'ID utilisateur défini sur le serveur LDAP doit avoir un ID utilisateur identique sur le système sur lequel IBM MQ est installé. Ces ID utilisateur doivent être dans le même cas, sinon le mappage entre les ID utilisateur peut échouer.
- Pour effectuer cette tâche, vous devez être un utilisateur disposant de privilèges suffisants pour éditer le fichier `mqwebuser.xml` :

– z/OS Sous z/OS, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` .

– Multi Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.

Procédure

1. Copiez l'exemple de fichier XML `ldap_registry.xml` à partir de l'un des chemins suivants:

   - ULW Sous UNIX, Linux, and Windows: *MQ_INSTALLATION_PATH* `/web/mq/samp/configuration`
   - z/OS Sous z/OS: `PathPrefix /web/mq/samp/configuration`

   où `PathPrefix` est le chemin d'installation de IBM MQ Unix System Services Components.
2. Placez l'exemple de fichier dans le répertoire approprié:

   - ULW

     Sous UNIX, Linux, and Windows : *MQ_DATA_PATH*`/web/installations/`*installationName*`/servers/mqweb`
   - z/OS

     Sous z/OS : *WLP_user_directory*`/servers/mqweb`

     où *WLP_user_directory* est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb** pour créer la définition de serveur mqweb.
3. Facultatif : Si vous avez modifié des paramètres de configuration dans `mqwebuser.xml`, copiez-les dans l'exemple de fichier.
4. Supprimez le fichier `mqwebuser.xml` existant et renommez l'exemple de fichier en `mqwebuser.xml`.
5. Editez le nouveau fichier `mqwebuser.xml` pour modifier les paramètres du registre LDAP dans les balises **ldapRegistry** et **idsLdapFilterProperties** .

   Pour plus d'informations sur la configuration des registres LDAP, voir Configuration des registres d'utilisateurs LDAP dans Liberty dans la documentation WebSphere Liberty .
6. Affectez des rôles aux utilisateurs et aux groupes en éditant le fichier `mqwebuser.xml` :

   Plusieurs rôles sont disponibles pour autoriser les utilisateurs et les groupes à utiliser le IBM MQ Consoleet le REST API. Chaque rôle accorde un niveau d'accès différent. Pour plus d'informations, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

   - Pour affecter des rôles et accorder l'accès au IBM MQ Console, ajoutez vos utilisateurs et groupes entre les balises **security-role** appropriées dans les balises **<enterpriseApplication id="com.ibm.mq.console">** .
   - Pour affecter des rôles et accorder l'accès au REST API, ajoutez vos utilisateurs et groupes entre les balises **security-role** appropriées dans les balises **<enterpriseApplication id="com.ibm.mq.rest">** .

Que faire ensuite

Choisissez comment les utilisateurs s'authentifient:

IBM MQ Consoleoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'intervalle d'expiration pour le jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.

- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

REST APIoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524. Vous pouvez configurer l'intervalle d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

z/OS V 9.1.0 Configuration d'un registre SAF pour IBM MQ Console et REST API

L'interface SAF (System Authorization Facility) permet au serveur mqweb d'appeler le gestionnaire de sécurité externe pour l'authentification et la vérification des autorisations. Un utilisateur peut ensuite se connecter à IBM MQ Console et à REST API avec un ID utilisateur et un mot de passe z/OS .

Avant de commencer

- Lorsque vous configurez un registre SAF, vous devez affecter un rôle aux utilisateurs. Chaque rôle fournit différents niveaux de privilèges pour accéder à IBM MQ Console et REST API, et détermine le contexte de sécurité qui est utilisé lorsqu'une opération autorisée est tentée. Vous devez comprendre ces rôles avant de configurer le registre. Pour plus d'informations sur chacun des rôles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.
- Vous avez besoin du processus WebSphere Liberty Angel en cours d'exécution pour utiliser l'interface autorisée pour SAF. Pour plus d'informations, voir Activation des services autorisés z/OS sur Liberty for z/OS .
- Pour effectuer cette tâche, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` et des droits permettant de définir des profils de gestionnaire de sécurité.

Remarque : V 9.1.0.20 Depuis IBM MQ 9.1.0 Fix Pack 20, l'exemple de fichier de configuration `zos_saf_registry.xml` a été mis à jour pour supprimer une entrée safAuthorization en double.

Cette mise à jour corrige un problème où une erreur ICH408I peut se produire lorsque MQ Console on z/OS est mis à niveau vers un niveau qui fournit WebSphere Liberty Profile 22.0.0.12 ou version ultérieure, c'est-à-dire à partir de IBM MQ 9.1.0 Fix Pack 15. La présence de plusieurs instructions safAuthorization n'est pas prise en charge et peut provoquer une erreur ICH408I lorsque des utilisateurs qui ne sont pas des rôles MQWebAdmin ou MQWebAdminRO, dans la classe EBJROLE, tentent d'accéder à un gestionnaire de files d'attente z/OS via MQ Console.

La valeur par défaut de **`racRouteLog`**, qui spécifie les types de tentatives d'accès à la journalisation, est `NONE`. Si vous avez besoin d'un rapport ou d'un enregistrement supplémentaire pour l'audit de sécurité, voir SAF Authorization (safAuthorization) pour plus d'informations.

Pourquoi et quand exécuter cette tâche

L'interface SAF permet au serveur mqweb d'appeler le gestionnaire de sécurité externe pour l'authentification et la vérification des autorisations pour IBM MQ Console et REST API.

Procédure

1. Suivez les étapes de la rubrique Activation des services autorisés z/OS sur Liberty for z/OS pour permettre à votre serveur mqweb d'accéder à l'utilisation des services autorisés z/OS .

   Un exemple de JCL permettant de démarrer le processus ange se trouve dans USS_ROOT/web/templates/zos/procs/bbgzangl.jcl, où USS_ROOT est le chemin d'accès dans les services système Unix où les composants IBM MQ for z/OS USS sont installés.

   Dans bbgzangl.jcl, modifiez l'instruction SET ROOT pour qu'elle pointe vers USS_ROOT/web, par exemple, /usr/lpp/mqm/V9R1M0/web.

   Pour plus d'informations sur l'arrêt et le démarrage du processus ange, voir Administration de Liberty sur z/OS .
2. Suivez les étapes de la rubrique Liberty: Configuration de l'utilisateur non authentifié SAF (System Authorization Facility) pour créer l'utilisateur non authentifié requis par Liberty.
3. Copiez le fichier zos_saf_registry.xml à partir du chemin suivant: PathPrefix /web/mq/samp/configuration où PathPrefix est le chemin d'installation d' IBM MQ Unix System Services Components.
4. Placez l'exemple de fichier dans le répertoire *WLP_user_directory*/servers/mqweb , où *WLP_user_directory* est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb** pour créer la définition de serveur mqweb.
5. Facultatif : Si vous avez précédemment modifié des paramètres de configuration dans mqwebuser.xml, copiez-les dans l'exemple de fichier.
6. Supprimez le fichier mqwebuser.xml existant et renommez l'exemple de fichier en mqwebuser.xml.
7. Personnalisez l'élément **safCredentials** dans mqwebuser.xml.
   a. Définissez **profilePrefix** sur un nom unique pour votre serveur Liberty. Si plusieurs serveurs mqweb s'exécutent sur un même système, vous devez choisir un nom différent pour chaque serveur, par exemple MQWEB910 et MQWEB905.
   b. Définissez **unauthenticatedUser** sur le nom de l'utilisateur non authentifié créé à l'étape «2», à la page 516.
8. Définissez l'ID application du serveur mqweb sur RACF.

   Le nom de ressource APPLID est la valeur que vous avez spécifiée dans l'attribut **profilePrefix** à l'étape «7», à la page 516. L'exemple suivant définit l'identificateur d'application du serveur mqweb dans RACF:

   ```
   RDEFINE APPL profilePrefix UACC(NONE)
   ```
9. Accordez à tous les utilisateurs, ou groupes, l'accès en lecture (READ) MQ Console ou REST API à l'identificateur d'application du serveur mqweb dans la classe APPL.

   Vous devez également effectuer cette opération pour l'utilisateur non authentifié défini à l'étape «2», à la page 516. L'exemple suivant accorde à un utilisateur un accès en lecture (READ) à l'identificateur d'application du serveur mqweb dans RACF:

   ```
   PERMIT profilePrefix CLASS(APPL) ACCESS(READ) ID(userID)
   ```
10. Définissez les profils dans la classe EJBROLE nécessaires pour permettre aux utilisateurs d'accéder aux rôles dans MQ Console et REST API.

L'exemple suivant définit les profils dans RACF, où **profilePrefix** est la valeur spécifiée pour l'attribut **profilePrefix** à l'étape «7», à la page 516.

```
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdminRO UACC(NONE)
```

11. Accordez aux utilisateurs l'accès aux rôles dans MQ Console et REST API.

    Pour ce faire, accordez aux utilisateurs ou aux groupes un accès en lecture (READ) à un ou plusieurs des profils de la classe EBJROLE créée à l'étape «10», à la page 516. Pour plus d'informations sur les rôles, voir «Rôles sur les IBM MQ Console et REST API», à la page 518.

    L'exemple suivant donne à un utilisateur l'accès au rôle MQWebAdmin pour le REST API dans RACF, où **profilePrefix** est la valeur spécifiée pour l'attribut **profilePrefix** à l'étape «7», à la page 516.

```
PERMIT profilePrefix.com.ibm.mq.rest.MQWebAdmin CLASS(EJBROLE) ACCESS(READ) ID(userID)
```

Résultats

Vous avez configuré l'authentification SAF pour IBM MQ Console et REST API.

Que faire ensuite

Choisissez comment les utilisateurs s'authentifient:

IBM MQ Consoleoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur entre un ID utilisateur et un mot de passe à l'écran de connexion IBM MQ Console . Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Aucune configuration supplémentaire n'est requise pour utiliser cette option d'authentification, mais vous pouvez éventuellement configurer l'intervalle d'expiration pour le jeton LTPA. Pour plus d'informations, voir Configuration de l'intervalle d'expiration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à IBM MQ Console, mais utilise le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

REST APIoptions d'authentification

- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir «Utilisation de l'authentification de base HTTP avec REST API», à la page 523.
- Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour plus d'informations, voir «Utilisation de l'authentification basée sur un jeton avec l'API REST», à la page 524. Vous pouvez configurer l'intervalle d'expiration du jeton LTPA. Pour plus d'informations, voir Configuration du jeton LTPA.
- Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à REST API, mais utilise

le certificat client à la place. Pour plus d'informations, voir «Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console», à la page 519.

V 9.1.0 Rôles sur les IBM MQ Console et REST API

Lorsque vous autorisez des utilisateurs et des groupes à utiliser IBM MQ Console ou REST API, vous devez affecter aux utilisateurs et aux groupes l'un des rôles disponibles: **MQWebAdmin**, **MQWebAdminRO**, **MQWebUser**, **MFTWebAdmin**et **MFTWebAdminRO**. Chaque rôle fournit différents niveaux de privilèges pour accéder à IBM MQ Console et REST API, et détermine le contexte de sécurité qui est utilisé lorsqu'une opération autorisée est tentée.

Remarque : A l'exception du rôle **MQWebUser** , l'ID utilisateur n'est pas sensible à la casse. Pour connaître les exigences spécifiques à ce rôle, voir «MQWebUser», à la page 518 .

MQWebAdmin
Un utilisateur ou un groupe auquel ce rôle est affecté peut effectuer toutes les opérations d'administration et fonctionne dans le contexte de sécurité de l'ID utilisateur du système d'exploitation utilisé pour démarrer le serveur mqweb.

Un utilisateur ou un groupe ayant ce rôle n'a pas accès aux services REST suivants:

- REST API pour MFT. Pour utiliser ces services, l'utilisateur ou le groupe doit également disposer du rôle **MFTWebAdmin** ou **MFTWebAdminRO** .
- messaging REST API. Pour utiliser le messaging REST API, le rôle **MQWebUser** doit être affecté à l'utilisateur.

MQWebAdminRO
Ce rôle permet d'accéder en lecture seule à IBM MQ Console ou REST API. Un utilisateur ou un groupe auquel ce rôle est affecté peut effectuer les opérations suivantes:

- Affichez et interrogez les opérations sur les objets IBM MQ tels que les files d'attente et les canaux.
- Parcourez les messages dans les files d'attente.

Un utilisateur ou un groupe auquel ce rôle est affecté fonctionne dans le contexte de sécurité de l'ID utilisateur du système d'exploitation utilisé pour démarrer le serveur mqweb.

Un utilisateur ou un groupe ayant ce rôle n'a pas accès aux services REST suivants:

- REST API pour MFT. Pour utiliser ces services, l'utilisateur ou le groupe doit également disposer du rôle **MFTWebAdmin** ou **MFTWebAdminRO** .
- messaging REST API. Pour utiliser le messaging REST API, le rôle **MQWebUser** doit être affecté à l'utilisateur.

MQWebUser
Un utilisateur ou un groupe auquel ce rôle est affecté peut effectuer toute opération que l'ID utilisateur est autorisé à effectuer sur le gestionnaire de files d'attente. Exemple :

- Opérations de démarrage et d'arrêt sur les objets IBM MQ tels que les canaux.
- Définissez et définissez des opérations sur des objets IBM MQ tels que des files d'attente et des canaux.
- Affichez et interrogez les opérations sur les objets IBM MQ tels que les files d'attente et les canaux.
- Insérez et extrayez des messages à l'aide de messaging REST API.

Un utilisateur ou un groupe auquel ce rôle est affecté agit dans le contexte de sécurité du principal et peut effectuer uniquement les opérations que l'ID utilisateur est autorisé à effectuer sur le gestionnaire de files d'attente.

Par conséquent, l'utilisateur ou le groupe défini dans le registre d'utilisateurs mqweb doit disposer des droits d'accès dans IBM MQ pour que cet utilisateur puisse effectuer des opérations. En utilisant ce rôle, vous pouvez contrôler finement quels utilisateurs ont quel type d'accès à des ressources IBM MQ spécifiques lorsqu'ils utilisent IBM MQ Console et REST API.

Remarque :

- La longueur maximale d'un ID utilisateur auquel ce rôle est affecté est de 12 caractères.
- La casse de l'ID utilisateur doit être la même dans le registre d'utilisateurs mqweb et sur le système IBM MQ . Si la casse de l'ID utilisateur est différente, l'utilisateur peut être authentifié par IBM MQ Console et REST API mais ne pas être autorisé à utiliser les ressources IBM MQ .

Un utilisateur ou un groupe ayant ce rôle n'a accès à aucun des services REST API for MFT . Pour utiliser ces services, l'utilisateur ou le groupe doit également disposer du rôle **MFTWebAdmin** ou **MFTWebAdminRO** .

MFTWebAdmin
Un utilisateur ou un groupe auquel ce rôle a été affecté peut effectuer toutes les opérations REST MFT et fonctionne dans le contexte de sécurité de l'ID utilisateur du système d'exploitation utilisé pour démarrer le serveur `mqweb` .

Un utilisateur ou un groupe ayant ce rôle n'a accès à aucun des services IBM MQ REST API .
Pour utiliser ces services, l'utilisateur ou le groupe doit également disposer du rôle **MQWebAdmin**, **MQWebAdminRO**ou **MQWebUser** .

MFTWebAdminRO
Ce rôle permet d'accéder en lecture seule à REST API for MFT . Un utilisateur ou un groupe auquel ce rôle est affecté peut effectuer des opérations en lecture seule (demandes GET) telles que le transfert de liste et les agents de liste.

Un utilisateur ou un groupe auquel ce rôle est affecté fonctionne dans le contexte de sécurité de l'ID utilisateur du système d'exploitation utilisé pour démarrer le serveur mqweb.

Un utilisateur ou un groupe ayant ce rôle n'a accès à aucun des services IBM MQ REST API .
Pour utiliser ces services, l'utilisateur ou le groupe doit également disposer du rôle **MQWebAdmin**, **MQWebAdminRO**ou **MQWebUser** .

Pour plus d'informations sur la configuration des utilisateurs et des groupes pour utiliser ces rôles, voir «Configuration des utilisateurs et des rôles», à la page 508.

Chevauchement de rôles

Plusieurs rôles peuvent être affectés à un utilisateur ou à un groupe. Lorsqu'un utilisateur effectue une opération dans cette situation, le rôle de privilège le plus élevé applicable à l'opération est utilisé. Par exemple, si un utilisateur disposant des rôles **MQWebAdminRO** et **MQWebUser** effectue une opération d'interrogation de file d'attente, le rôle **MQWebAdminRO** est utilisé et l'opération est tentée dans le contexte de l'ID utilisateur système qui a démarré le serveur Web. Si ce même utilisateur effectue une opération de définition, le rôle **MQWebUser** est utilisé et l'opération est tentée dans le contexte du principal.

ULW V 9.1.0 Utilisation de l'authentification par certificat client avec REST API et IBM MQ Console

Vous pouvez mapper des certificats client à des principaux pour authentifier les utilisateurs IBM MQ Console et REST API .

Avant de commencer

- Configurez les utilisateurs, les groupes et les rôles pour qu'ils soient autorisés à utiliser IBM MQ Console et REST API. Pour plus d'informations, voir «Configuration des utilisateurs et des rôles», à la page 508.
- Lorsque vous utilisez le REST API, vous pouvez interroger les données d'identification de l'utilisateur en cours à l'aide de la méthode HTTP GET sur la ressource `login` , en fournissant le certificat client pour authentifier la demande. Cette demande renvoie des informations sur le nom d'utilisateur et les rôles affectés à l'utilisateur. Pour plus d'informations, voir GET `/login`.
- Lorsque vous mappez des certificats client à des principaux pour authentifier les utilisateurs, le nom distinctif du certificat client est utilisé pour établir une correspondance avec les utilisateurs du registre d'utilisateurs configuré:

- Pour un registre de base, le nom usuel (CN) est comparé à l'utilisateur. Par exemple, `CN=Fred, O=IBM, C=GB` est mis en correspondance avec le nom d'utilisateur `Fred`.
- Pour un registre LDAP, par défaut, le nom distinctif complet est comparé à LDAP. Vous pouvez configurer des filtres et des mappages pour personnaliser la mise en correspondance. Pour plus d'informations, voir Liberty :LDAP certificate map mode dans la documentation WebSphere Liberty .

Pourquoi et quand exécuter cette tâche

Lorsqu'un utilisateur s'authentifie à l'aide d'un certificat client, le certificat est utilisé à la place d'un nom d'utilisateur et d'un mot de passe. Pour le REST API, le certificat client est fourni avec chaque demande REST pour authentifier l'utilisateur. Pour IBM MQ Console, lorsqu'un utilisateur se connecte avec un certificat, il ne peut pas être déconnecté.

La procédure suppose les informations suivantes:

- Votre fichier `mqwebuser.xml` est basé sur l'un des exemples suivants:
  - `basic_registry.xml`
  - `local_os_registry.xml`
  - `ldap_registry.xml`
- que vous utilisez un système UNIX, Linuxou Windows .
- Vous êtes un utilisateur privilégié.

Pour configurer l'authentification par certificat client avec un fichier de clés RACF sous z/OS, suivez la procédure décrite dans «Configuration de TLS pour REST API et IBM MQ Console sous z/OS», à la page 532.

Remarque : La procédure suivante décrit les étapes nécessaires à l'utilisation des certificats client avec IBM MQ Console et REST API. Pour des raisons de commodité pour les développeurs, les étapes expliquent comment créer et utiliser des certificats autosignés. Toutefois, pour la production, utilisez des certificats obtenus auprès d'une autorité de certification.

Procédure

1. Démarrez le serveur mqweb en entrant la commande **strmqweb** sur la ligne de commande.
2. Créez un certificat client:
   a) Créez un magasin de clés PKCS#12 :
      i) Ouvrez l'outil IBM Key Management en entrant la commande **strmqikm** sur la ligne de commande.
      ii) Dans le menu **Fichier de base de données de clés** de l'outil IBM Key Management, cliquez sur **Nouveau**.
      iii) Sélectionnez **PKCS12** dans la liste **Type de base de données de clés** .
      iv) Sélectionnez un emplacement pour enregistrer le magasin de clés et entrez un nom approprié dans la zone **Nom de fichier** . Exemple : `user.p12`
      v) Définissez un mot de passe lorsque vous y êtes invité.
   b) Créez le certificat, soit en créant un certificat autosigné, soit en obtenant un certificat auprès d'une autorité de certification:
      - Créez un certificat autosigné:
        i) Cliquez sur **New Self-Signed (nouveau auto-signé)**.
        ii) Entrez `user` dans la zone **Key Label** .
        iii) Si vous utilisez un registre d'utilisateurs de base, entrez le nom d'un utilisateur de votre registre d'utilisateurs dans la zone **Nom usuel** . Par exemple, `mqadmin`. Pour un registre d'utilisateurs LDAP, assurez-vous que le nom distinctif du certificat correspond au nom distinctif dans le registre LDAP.

iv) Cliquez sur **OK**.

- Obtenir un certificat auprès d'une autorité de certification. Le certificat de l'autorité de certification doit inclure le nom d'utilisateur approprié dans le nom usuel (CN) de la zone du nom distinctif (DN):

  i) Demandez un nouveau certificat. Dans le menu **Créer** , cliquez sur **Nouvelle demande de certificat**.

  ii) Dans la zone **Key Label** , entrez le libellé du certificat.

  iii) Si vous utilisez un registre d'utilisateurs de base, dans la zone **Nom usuel** , entrez le nom de l'utilisateur auquel le certificat est destiné.

  Si vous utilisez un registre de système d'exploitation local, la zone **Nom usuel** doit correspondre à l'ID utilisateur du système d'exploitation local.

  Pour un registre d'utilisateurs LDAP, assurez-vous que le nom distinctif du certificat correspond au nom distinctif dans le registre LDAP.

  iv) Entrez ou sélectionnez des valeurs pour les zones restantes, selon le cas.

  v) Choisissez l'emplacement d'enregistrement de la demande de certificat et le nom de fichier de la demande de certificat, puis cliquez sur **OK**.

  vi) Envoyez le fichier de demande de certificat à une autorité de certification.

  vii) Lorsque vous disposez du certificat de l'autorité de certification, ouvrez l'outil IBM Key Management en entrant la commande **strmqikm** sur la ligne de commande.

  viii) Dans le menu **Fichier de base de données de clés** de l'outil IBM Key Management, cliquez sur **Ouvrir**.

  ix) Sélectionnez le magasin de clés PKCS#12 qui contient le certificat client. Exemple: `user.p12`

  x) Cliquez sur **Recevoir**, sélectionnez le certificat approprié, puis cliquez sur **OK**.

3. Extrayez la partie publique du certificat client:
   a) Ouvrez l'outil IBM Key Management en entrant la commande **strmqikm** sur la ligne de commande.
   b) Dans le menu **Fichier de base de données de clés** de l'outil IBM Key Management, cliquez sur **Ouvrir**.
   c) Sélectionnez le magasin de clés PKCS#12 qui contient le certificat client. Exemple: `user.p12`
   d) Sélectionnez le certificat client dans la liste des certificats de l'outil IBM Key Management.
   e) Cliquez sur **Extraire le certificat**.
   f) Sélectionnez un emplacement pour enregistrer le certificat et entrez un nom de fichier approprié dans la zone **Nom du fichier de certificat** . Par exemple, `user.arm`.
4. Importez la partie publique du certificat client dans le magasin de clés de confiance du serveur mqweb en tant que certificat de signataire afin que le serveur puisse valider le certificat client:
   a) Créez un magasin de clés `trust.jks` à utiliser par le serveur mqweb, s'il n'en existe pas déjà un:
      i) Dans le menu **Fichier de base de données de clés** de l'outil IBM Key Management, cliquez sur **Nouveau**.
      ii) Sélectionnez **JKS** dans la liste **Key database type**.
      iii) Cliquez sur **Parcourir** et accédez à: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb/resources/security`.

      Ce répertoire doit déjà contenir un fichier `key.jks` . Si un fichier `trust.jks` existe déjà, ouvrez le fichier existant au lieu de le remplacer.

      iv) Entrez `trust.jks` dans la zone **Nom de fichier** .
      v) Définissez un mot de passe lorsque vous y êtes invité.
   b) Dans le menu déroulant, sélectionnez **Signer Certificates**.
   c) Cliquez sur **Ajouter**.

d) Sélectionnez le fichier de bras approprié et cliquez sur **OK**. Par exemple, sélectionnez `user.arm`.

e) Entrez un libellé pour le certificat.

5. Modifiez le mot de passe du magasin de clés du serveur mqweb:

a) Dans le menu **Key Database File**, cliquez sur **Ouvrir**.

b) Sélectionnez **JKS** dans la liste **Key database type**.

c) Cliquez sur **Parcourir** et accédez à *MQ_DATA_PATH*/web/installations/*installationName*/servers/mqweb/resources/security

d) Sélectionnez le magasin de clés `key.jks` et cliquez sur **Ouvrir**.

e) A l'invite, entrez le mot de passe. Le mot de passe par défaut est `password`.

f) Dans le menu **Fichier de base de données de clés** , cliquez sur **Modifier le mot de passe**.

g) Entrez un nouveau mot de passe pour le magasin de clés.

6. Activez l'authentification par certificat client dans le fichier `mqwebuser.xml` :

Le fichier `mqwebuser.xml` se trouve dans le chemin suivant: *MQ_DATA_PATH*/web/installations/*installationName*/servers/mqweb

a) Supprimez la mise en commentaire de la section du fichier `mqwebuser.xml` qui active l'authentification par certificat client. La section contient le texte suivant:

```
<keyStore id="defaultKeyStore" location="key.jks" type="JKS" password="password"/>
    <keyStore id="defaultTrustStore" location="trust.jks" type="JKS" password="password"/>
    <ssl id="thisSSLConfig" clientAuthenticationSupported="true"
keyStoreRef="defaultKeyStore"
           trustStoreRef="defaultTrustStore" sslProtocol="TLSv1.2"
serverKeyAlias="default"/>
    <sslDefault sslRef="thisSSLConfig"/>
```

b) Vérifiez que la valeur **serverKeyAlias** correspond au nom du certificat serveur. Si vous utilisez le certificat serveur par défaut, la valeur est correcte.

c) Remplacez la valeur de **password** pour `defaultKeyStore` par une version codée du mot de passe du magasin de clés `key.jks` :

i) Dans le répertoire *MQ_INSTALLATION_PATH*/web/bin , entrez la commande suivante sur la ligne de commande:

```
securityUtility encode password
```

ii) Placez la sortie de cette commande dans la zone **password** de `defaultKeyStore`.

d) Modifiez la valeur de **password** pour `defaultTrustStore` afin qu'elle corresponde au mot de passe du magasin de clés `trust.jks` :

i) Dans le répertoire *MQ_INSTALLATION_PATH*/web/bin , entrez la commande suivante sur la ligne de commande:

```
securityUtility encode password
```

ii) Placez la sortie de cette commande dans la zone **password** de `defaultTrustStore`.

e) Supprimez ou mettez en commentaire la ligne suivante du fichier `mqwebuser.xml` :

```
<sslDefault sslRef="mqDefaultSSLConfig"/>
```

7. Arrêtez le serveur mqweb en entrant la commande **endmqweb** sur la ligne de commande.

8. Démarrez le serveur mqweb en entrant la commande **strmqweb** sur la ligne de commande.

9. Utilisez le certificat client pour l'authentification:

- Pour utiliser le certificat client avec IBM MQ Console, installez le certificat client dans le navigateur Web utilisé pour accéder au IBM MQ Console. Par exemple, installez le certificat client `user.p12` en tant que certificat personnel.

- Pour utiliser le certificat client avec REST API, indiquez le certificat client avec chaque demande REST. Lorsque vous utilisez les méthodes HTTP POST, PATCH ou DELETE, vous devez fournir une authentification supplémentaire avec le certificat client pour empêcher les attaques de falsification de requêtes entre sites. Autrement dit, l'authentification supplémentaire est utilisée pour confirmer que les données d'identification utilisées pour authentifier la demande sont utilisées par le propriétaire des données d'identification.

  Cette authentification supplémentaire est fournie par l'en-tête HTTP `ibm-mq-rest-csrf-token` . Définissez la valeur de l'en-tête `ibm-mq-csrf-token` sur n'importe quoi, y compris un blanc, puis soumettez la demande.

Exemple

Important : Dans l'exemple, toutes les implémentations cURL ne prenant pas en charge les certificats autosignés, vous devez utiliser une implémentation cURL .

L'exemple cURL suivant montre comment créer une file d'attente `Q1`, sur le gestionnaire de files d'attente `QM1`, avec authentification par certificat client. La configuration exacte de cette commande cURL dépend des bibliothèques sur lesquelles cURL a été généré. L'exemple est basé sur un système Windows , avec cURL généré sur OpenSSL.

- Utilisez la méthode HTTP POST avec la ressource de file d'attente, en vous authentifiant avec le certificat client et en incluant l'en-tête HTTP `ibm-mq-rest-csrf-token` avec une valeur arbitraire. Cette valeur peut être n'importe quoi, y compris vide. L'indicateur `--cert-type` spécifie que le certificat est un certificat PKCS#12 . L'indicateur `--cert` spécifie l'emplacement du certificat, suivi d'un signe deux-points (:), puis du mot de passe du certificat:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -
-cert-type P12 --cert c:\user.p12:password
-H "ibm-mq-rest-csrf-token: value"
-H "Content-Type: application/json" --data "{\"name\":\"Q1\"}"
```

V 9.1.0 Utilisation de l'authentification de base HTTP avec REST API

Les utilisateurs du REST API peuvent s'authentifier en fournissant leur ID utilisateur et leur mot de passe dans un en-tête HTTP. Pour utiliser cette méthode d'authentification avec des méthodes HTTP, telles que POST, PATCH et DELETE, l'en-tête HTTP `ibm-mq-rest-csrf-token` doit également être fourni, ainsi qu'un ID utilisateur et un mot de passe.

Avant de commencer

- Configurez les utilisateurs, les groupes et les rôles pour qu'ils soient autorisés à utiliser le REST API. Pour plus d'informations, voir «Configuration des utilisateurs et des rôles», à la page 508.
- Vérifiez que l'authentification de base HTTP est activée. Vérifiez que le code XML suivant est présent et qu'il n'est pas mis en commentaire dans le fichier `mqwebuser.xml` . Ce code XML doit se trouver dans les balises `<featureManager>` :

```
<feature>basicAuthenticationMQ-1.0</feature>
```

z/OS Sous z/OS, vous devez être un utilisateur disposant d'un accès en écriture à `mqwebuser.xml` pour pouvoir éditer ce fichier.

Multi Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié pour pouvoir éditer le fichier `mqwebuser.xml` .
- Vérifiez que vous utilisez une connexion sécurisée lorsque vous envoyez des demandes REST. Comme la combinaison du nom d'utilisateur et du mot de passe est codée, mais pas chiffrée, vous devez utiliser une connexion sécurisée (HTTPS) lorsque vous utilisez l'authentification de base HTTP avec REST API.
- Vous pouvez interroger les données d'identification de l'utilisateur en cours à l'aide de la méthode HTTP GET sur la ressource `login` , en fournissant les informations d'authentification de base pour

authentifier la demande. Cette demande renvoie des informations sur le nom d'utilisateur et les rôles affectés à l'utilisateur. Pour plus d'informations, voir GET /login.

Procédure

1. Concaténez le nom d'utilisateur avec un signe deux-points et le mot de passe. Notez que le nom d'utilisateur est sensible à la casse.

   Par exemple, le nom d'utilisateur admin et le mot de passe admin deviennent la chaîne suivante:

   ```
   admin:admin
   ```

2. Codez ce nom d'utilisateur et cette chaîne de mot de passe dans le codage base64 .
3. Incluez ce nom d'utilisateur et ce mot de passe codés dans un en-tête HTTP `Authorization: Basic` .

   Par exemple, avec le nom d'utilisateur codé admin et le mot de passe admin, l'en-tête suivant est créé:

   ```
   Authorization: Basic YWRtaW46YWRtaW4=
   ```

4. Lorsque vous utilisez les méthodes HTTP POST, PATCH ou DELETE, vous devez fournir une authentification supplémentaire, ainsi qu'un nom d'utilisateur et un mot de passe.

   Cette authentification supplémentaire est fournie par l'en-tête HTTP `ibm-mq-rest-csrf-token` . L'en-tête HTTP `ibm-mq-rest-csrf-token` doit être présent dans la demande, mais sa valeur peut être quelconque, y compris vide.
5. Soumettez votre demande REST à IBM MQ avec les en-têtes appropriés.

Exemple

L'exemple suivant montre comment créer une nouvelle file d'attente Q1, sur le gestionnaire de files d'attente QM1, avec authentification de base, sur les systèmes Windows . L'exemple utilise cURL:

- Utilisez la méthode HTTP POST avec la ressource de file d'attente, en vous authentifiant avec l'authentification de base et en incluant l'en-tête HTTP `ibm-mq-rest-csrf-token` avec une valeur arbitraire. Cette valeur peut être n'importe quoi, y compris vide:

  ```
  curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST
  -u mqadmin:mqadmin
  -H "ibm-mq-rest-csrf-token: value"
  -H "Content-Type: application/json" --data "{\"name\":\"Q1\"}"
  ```

V 9.1.0 Utilisation de l'authentification basée sur un jeton avec l'API REST

Les utilisateurs de REST API peuvent s'authentifier en fournissant un ID utilisateur et un mot de passe à la ressource REST API `login` à l'aide de la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur d'authentifier les demandes futures. Ce jeton LTPA a le préfixe `LtpaToken2`. L'utilisateur peut se déconnecter à l'aide de la méthode HTTP DELETE et peut interroger les informations de connexion de l'utilisateur en cours à l'aide de la méthode HTTP GET.

Avant de commencer

- Configurez les utilisateurs, les groupes et les rôles pour qu'ils soient autorisés à utiliser le REST API. Pour plus d'informations, voir «Configuration des utilisateurs et des rôles», à la page 508.
- Par défaut, le nom du cookie qui inclut le jeton LTPA commence par `LtpaToken2`et inclut un suffixe qui peut être modifié lorsque le serveur mqweb est redémarré. Ce nom de cookie aléatoire permet à plusieurs serveurs mqweb de s'exécuter sur le même système. Toutefois, si vous souhaitez que le nom du cookie reste cohérent, vous pouvez spécifier le nom du cookie à l'aide de la commande **setmqweb** . Pour plus d'informations, voir Configuration du jeton LTPA.

- Par défaut, le cookie de jeton LTPA expire au bout de 120 minutes. Vous pouvez configurer l'heure d'expiration du cookie de jeton LTPA à l'aide de la commande **setmqweb** . Pour plus d'informations, voir Configuration du jeton LTPA.
- Vérifiez que vous utilisez une connexion sécurisée lorsque vous envoyez des demandes REST. Lorsque vous utilisez la méthode HTTP POST sur la ressource `login` , la combinaison de nom d'utilisateur et de mot de passe envoyée avec la demande n'est pas chiffrée. Par conséquent, vous devez utiliser une connexion sécurisée (HTTPS) lorsque vous utilisez l'authentification basée sur un jeton avec REST API. Par défaut, vous ne pouvez pas utiliser HTTP avec l'authentification par jeton LTPA. Vous pouvez activer le jeton LTPA à utiliser par les connexions HTTP non sécurisées en définissant **secureLTPA** sur `False`. Pour plus d'informations, voir Configuration du jeton LTPA.
- Vous pouvez interroger les données d'identification de l'utilisateur en cours à l'aide de la méthode HTTP GET sur la ressource `login` , en fournissant le jeton LTPA pour authentifier la demande. Cette demande renvoie des informations sur le nom d'utilisateur et les rôles affectés à l'utilisateur. Pour plus d'informations, voir GET `/login`.

Procédure

1. Connectez-vous à un utilisateur:
   a) Utilisez la méthode HTTP POST sur la ressource `login` :

   ```
   https://host:port/ibmmq/rest/v1/login
   ```

   Incluez le nom d'utilisateur et le mot de passe dans le corps de la demande JSON, au format suivant:

   ```
   {
       "username" : name,
       "password" : password
   }
   ```

   b) Stockez le jeton LTPA renvoyé par la demande dans le magasin de cookies local. Par défaut, ce jeton LTPA a le préfixe `LtpaToken2`.
2. Authentifiez les demandes REST avec le jeton LTPA stocké en tant que cookie avec chaque demande.
   Pour les demandes qui utilisent les méthodes HTTP PUT, PATCH ou DELETE, incluez un en-tête `ibm-mq-rest-csrf-token` . La valeur de cet en-tête peut être n'importe quoi, y compris vide.
3. Déconnecter un utilisateur:
   a) Utilisez la méthode HTTP DELETE sur la ressource `login` :

   ```
   https://host:9443/ibmmq/rest/v1/login
   ```

   Vous devez fournir le jeton LTPA en tant que cookie pour authentifier la demande et inclure un en-tête `ibm-mq-rest-csrf-token` . La valeur de cet en-tête peut être n'importe quoi, y compris vide

   b) Traitez l'instruction de suppression du jeton LTPA du magasin de cookies local.

   Remarque : Si l'instruction n'est pas traitée et que le jeton LTPA reste dans le magasin de cookies local, le jeton LTPA peut être utilisé pour authentifier les futures demandes REST. C'est-à-dire que lorsque l'utilisateur tente de s'authentifier avec le jeton LTPA après la fin de la session, une nouvelle session est créée qui utilise le jeton existant.

Exemple

L'exemple cURL suivant montre comment créer une nouvelle file d'attente Q1, sur le gestionnaire de files d'attente QM1, avec authentification basée sur un jeton, sur les systèmes Windows :

- Connectez-vous et ajoutez le jeton LTPA avec le préfixe `LtpaToken2`au magasin de cookies local. Les informations de nom d'utilisateur et de mot de passe sont incluses dans le corps JSON. L'indicateur `-c` spécifie l'emplacement du fichier dans lequel stocker le jeton:

```
curl -k https://localhost:9443/ibmmq/rest/v1/login -X POST
-H "Content-Type: application/json" --data
"{\"username\":\"mqadmin\",\"password\":\"mqadmin\"}"
-c c:\cookiejar.txt
```

- Créez une file d'attente. Utilisez la méthode HTTP POST avec la ressource de file d'attente, en vous authentifiant avec le jeton LTPA. Le jeton LTPA avec le préfixe `LtpaToken2` est extrait du fichier `cookiejar.txt` à l'aide de l'indicateur `-b` . La protection CSRF est assurée par la présence de l'en-tête HTTP `ibm-mq-rest-csrf-token` :

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -b
c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json"
--data "{\"name\":\"Q1\"}"
```

- Déconnectez-vous et supprimez le jeton LTPA du magasin de cookies local. Le jeton LTPA est extrait du fichier `cookiejar.txt` à l'aide de l'indicateur `-b` . La protection CSRF est assurée par la présence de l'en-tête HTTP `ibm-mq-rest-csrf-token` . L'emplacement du fichier `cookiejar.txt` est spécifié par l'indicateur `-c` afin que le jeton LTPA soit supprimé du fichier:

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X DELETE
-H "ibm-mq-rest-csrf-token: value" -b c:\cookiejar.txt
-c c:\cookiejar.txt
```

Référence associée

POST /login

GET/login

Supprimer /login

V 9.1.3 Incorporation d'IBM MQ Console dans une trame d'information

L'élément HTML < iframe> peut être utilisé pour imbriquer une page Web dans une autre à l'aide d'un cadre en ligne (IFrame). Pour des raisons de sécurité, le IBM MQ Console ne peut pas être imbriqué dans un IFrame par défaut. Toutefois, vous pouvez activer un IFrame à l'aide de la propriété de configuration **mqConsoleFrameAncestors** sur le serveur mqweb.

Pourquoi et quand exécuter cette tâche

Le serveur mqweb gère une liste autorisée des origines des pages Web qui peuvent intégrer le IBM MQ Console à l'aide d'un IFrame. Une origine est une combinaison d'un schéma d'URL, d'un domaine et d'un port, par exemple, `https://example.com:1234`.

Vous pouvez utiliser la propriété de configuration **mqConsoleFrameAncestors** sur le serveur mqweb pour spécifier les entrées dans la liste.

Par défaut, **mqConsoleFrameAncestors** est vide, ce qui signifie que IBM MQ Console ne peut pas être imbriqué dans un IFrame.

Procédure

Spécifiez une liste d'origines de pages Web, qui peuvent imbriquer le IBM MQ Console dans un IFrame, en entrant la commande suivante:

```
setmqweb properties -k mqConsoleFrameAncestors -v allowedOrigins
```

où *allowedOrigins* est une liste d'origines séparées par des virgules. Chaque origine doit se composer des éléments suivants:

- Un nom d'hôte ou une adresse IP

- Un schéma d'URL facultatif
- Numéro de port facultatif

Notez que le nom d'hôte peut commencer par le caractère générique (*) et que le numéro de port peut également utiliser le caractère générique (*).

Exemples d'origines:

```
https://example.com:1234
```

qui permet à n'importe quelle page Web servie à partir de `https://example.com:1234` d'imbriquer le IBM MQ Console dans un IFrame.

```
https://*.example.com:*
```

qui permet à toute page Web HTTPS avec un nom d'hôte se terminant par `example.com`, et utilisant n'importe quel port, d'imbriquer le IBM MQ Console dans un IFrame.

Exemple

L'exemple suivant permet à IBM MQ Console d'être imbriqué dans un IFrame à partir de pages Web servies à partir de `https://site2.example.com:1234` ou de `https://site2.example.com:1235`:

```
setmqweb properties -k mqConsoleFrameAncestors -v
https://site2.example.com:1234,https://site2.example.com:1235
```

V 9.1.0 Configuration de CORS pour REST API

Par défaut, un navigateur Web n'autorise pas les scripts, tels que JavaScript, à appeler REST API lorsque le script n'est pas de la même origine que le REST API. C'est-à-dire que les demandes d'origine croisée ne sont pas activées. Vous pouvez configurer le partage de ressources d'origine croisée (CORS) pour autoriser les demandes d'origine croisée provenant d'origines spécifiées.

Pourquoi et quand exécuter cette tâche

Vous pouvez accéder à REST API via un navigateur Web, par exemple via un script. Comme ces demandes proviennent d'une origine différente de celle de REST API, le navigateur Web refuse la demande car il s'agit d'une demande d'origine croisée. L'origine est différente si le domaine, le port ou le schéma n'est pas le même.

Par exemple, si vous disposez d'un script hébergé sur `http://localhost:1999/` , vous effectuez une demande inter-origine si vous émettez une requête HTTP GET sur un site Web hébergé sur `https://localhost:9443/`. Cette demande est une demande inter-origine car les numéros de port et le schéma (HTTP) sont différents.

Vous pouvez activer les demandes inter-origines en configurant CORS et en spécifiant les origines qui sont autorisées à accéder à REST API.

Pour plus d'informations sur CORS, voir https://www.w3.org/TR/cors/ et https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS.

Procédure

1. Affichez la configuration en cours en entrant la commande suivante:

   ```
   dspmqweb properties -a
   ```

   L'entrée `mqRestCorsAllowedOrigins` indique les origines autorisées. L'entrée `mqRestCorsMaxAgeInSeconds` indique la durée, en secondes, pendant laquelle le navigateur Web peut mettre en cache les résultats des vérifications préalables à la mise en cache CORS.

2. Indiquez les origines autorisées à accéder à REST API en entrant la commande suivante:

   `setmqweb properties -k mqRestCorsAllowedOrigins -v allowedOrigins`

   où *allowedOrigins* indique l'origine à partir de laquelle vous souhaitez autoriser les demandes inter-origine. Vous pouvez utiliser un astérisque entre guillemets, "*", pour autoriser toutes les demandes d'origine croisée. Vous pouvez entrer plusieurs origines dans une liste séparée par des virgules, en les plaçant entre guillemets. Pour n'autoriser aucune demande d'origine croisée, entrez des guillemets vides comme valeur pour *allowedOrigins*.
3. Indiquez la durée, en secondes, pendant laquelle vous souhaitez autoriser un navigateur Web à mettre en cache les résultats des vérifications CORS préalables à la mise en cache en entrant la commande suivante:

   `setmqweb properties -k mqRestCorsMaxAgeInSeconds -v time`

Exemple

L'exemple suivant illustre les demandes d'origine croisée activées pour `http://localhost:9883`, `https://localhost:1999`et `https://localhost:9663`. L'âge maximal des résultats mis en cache des vérifications CORS avant vol est défini sur 90 secondes:

```
setmqweb properties -k mqRestCorsAllowedOrigins -v "http://localhost:9883,https://
localhost:1999,https://localhost:9663"
setmqweb properties -k mqRestCorsMaxAgeInSeconds -v 90
```

Configuration de la validation de l'en-tête d'hôte pour IBM MQ Console et REST API

Vous pouvez configurer le serveur mqweb pour restreindre l'accès à IBM MQ Console et à REST API de sorte que seules les demandes envoyées avec un en-tête d'hôte correspondant à une liste blanche spécifiée soient traitées. Une erreur est renvoyée si une valeur d'en-tête d'hôte qui n'est pas dans la liste blanche est utilisée.

Pourquoi et quand exécuter cette tâche

Le serveur mqweb utilise des hôtes virtuels pour définir la liste autorisée des en-têtes d'hôte acceptables. Pour plus d'informations sur les hôtes virtuels, voir la documentation WebSphere Liberty : https://www.ibm.com/docs/SSEQTP_liberty/com.ibm.websphere.wlp.doc/ae/cwlp_virtual_hosts.html

Pour effectuer cette tâche, vous devez être un utilisateur disposant de privilèges suffisants pour éditer le fichier `mqwebuser.xml` :

- z/OS Sous z/OS, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` .
- Multi Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.

Procédure

1. Ouvrez le fichier `mqwebuser.xml`. Ce fichier se trouve dans l'un des emplacements suivants:
   - ULW

     Sous UNIX, Linux, and Windows : *MQ_DATA_PATH*`/web/installations/`*installationName*`/servers/mqweb`
   - z/OS

     Sous z/OS : *WLP_user_directory*`/servers/mqweb`

     où *WLP_user_directory* est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb** pour créer la définition de serveur mqweb.
2. Ajoutez ou supprimez la mise en commentaire du code suivant dans le fichier `mqwebuser.xml` :

```
<virtualHost allowFromEndpointRef="defaultHttpEndpoint" id="default_host">
        <hostAlias>localhost:9080</hostAlias>
</virtualHost>
```

3. Editez la zone **<hostAlias>** en insérant la combinaison de nom d'hôte et de port que vous souhaitez autoriser.

   Cette combinaison peut être le nom d'hôte et le nom de port que vous avez utilisés dans la configuration du serveur mqweb. Par exemple, si vous utilisez la configuration par défaut de `localhost:9443`, vous pouvez utiliser `localhost:9443` dans la zone **<hostAlias>** .

   Si nécessaire, vous pouvez ajouter plusieurs zones **<hostAlias>** dans les balises **<virtualHost>** pour autoriser davantage de combinaisons de nom d'hôte et de port. Par exemple, pour autoriser les en-têtes d'hôte qui utilisent un port HTTP ainsi que les en-têtes d'hôte qui utilisent le port HTTPS.

V 9.1.0 Audit

Les enregistrements d'audit des opérations effectuées dans IBM MQ Console et REST API peuvent être générés en activant les événements de commande et de configuration du gestionnaire de files d'attente, et sur UNIX, Linux, and Windows les modifications d'état significatives sont enregistrées dans les fichiers journaux du serveur mqweb.

Changements d'état significatifs

ULW

Sous UNIX, Linux, and Windows, IBM MQ Console enregistre les changements d'état significatifs sous forme de messages dans les journaux du serveur mqweb. Chaque message indique le nom du principal authentifié qui a demandé l'opération.

Les modifications d'état importantes, telles que la création, le démarrage, l'arrêt ou la suppression des gestionnaires de files d'attente, sont consignées dans les fichiers `messages.log` et `console.log` du serveur mqweb au niveau de journalisation [AUDIT]. Chaque entrée de journal indique le nom du principal authentifié qui a demandé l'opération.

Les fichiers `messages.log` et `console.log` se trouvent à l'emplacement suivant:

- ULW Sous UNIX, Linux, and Windows : *MQ_DATA_PATH*`\web\installations\`*installationName*`\servers\mqweb\logs`

Pour plus d'informations sur la configuration des niveaux de journalisation du serveur mqweb, voir Configuration de la journalisation.

Evénements de commande et de configuration

Vous pouvez éventuellement activer des événements de commande et de configuration sur le gestionnaire de files d'attente pour fournir des informations sur la plupart des activités IBM MQ Console et REST API . Par exemple, la création de canaux et l'interrogation de files d'attente génèrent des événements de commande et de configuration. Pour plus d'informations sur l'activation des événements de commande et de configuration, voir Contrôle des événements de configuration, de commande et de consignateur.

Pour ces messages d'événement de commande et de configuration, la zone MQIACF_EVENT_ORIGIN est définie sur MQEVO_REST et la zone MQCACF_EVENT_APPL_IDENTITY indique les 32 premiers caractères du nom principal authentifié. Si un utilisateur a le rôle **MQWebAdmin** ou **MQWebAdminRO** , la zone MQCACF_EVENT_USER_ID indique l'ID utilisateur du serveur mqweb, et non le nom d'utilisateur du principal qui a émis la commande. Toutefois, si l'utilisateur possède le rôle **MQWebUser** , MQCACF_EVENT_USER_ID indique le nom d'utilisateur du principal qui a émis la commande.

Concepts associés

«Audit», à la page 472

Vous pouvez vérifier les intrusions de sécurité ou les tentatives d'intrusion à l'aide de messages d'événement. Vous pouvez également vérifier la sécurité de votre système à l'aide de la IBM MQ Explorer.

Remarques relatives à la sécurité pour IBM MQ Console et REST API sur z/OS

Les IBM MQ Console et REST API disposent de fonctions de sécurité contrôlant si un utilisateur peut émettre, afficher ou modifier des commandes. Les commandes sont ensuite transmises au gestionnaire de files d'attente, puis la sécurité du gestionnaire de files d'attente est utilisée pour contrôler si l'utilisateur est autorisé à exécuter la commande sur ce gestionnaire de files d'attente spécifique.

Procédure

1. Vérifiez que l'ID utilisateur de la tâche démarrée du serveur mqweb dispose des droits appropriés pour émettre certaines commandes PCF et accéder à certaines files d'attente. Pour plus d'informations, voir «Droits requis par l'ID utilisateur de la tâche démarrée du serveur mqweb», à la page 530.
2. Vérifiez que tous les utilisateurs auxquels le rôle `MQWebUser` a été attribué disposent des droits appropriés.

   Les utilisateurs IBM MQ Console et REST API affectés au rôle `MQWebUser` fonctionnent dans le contexte de sécurité du principal. Ces ID utilisateur peuvent uniquement effectuer des opérations que l'ID utilisateur est autorisé à effectuer sur le gestionnaire de files d'attente et doivent être autorisés à accéder aux mêmes files d'attente système que l'espace adresse du serveur mqweb.

   L'ID utilisateur de la tâche démarrée du serveur mqweb doit disposer d'un autre accès utilisateur à tous les utilisateurs affectés au rôle `MQWebUser` .

   Pour plus d'informations sur l'octroi des droits appropriés aux utilisateurs ayant le rôle `MQWebUser` , voir «Accès aux ressources IBM MQ requises pour utiliser MQ Console ou REST API», à la page 531.
3. Facultatif : Configurez TLS pour IBM MQ Console et REST API. Pour plus d'informations, voir «Configuration de TLS pour REST API et IBM MQ Console sous z/OS», à la page 532.

z/OS Droits requis par l'ID utilisateur de la tâche démarrée du serveur mqweb

Sous z/OS, l'ID utilisateur de la tâche démarrée du serveur mqweb requiert certains droits pour émettre des commandes PCF et accéder aux ressources système.

L'ID utilisateur de la tâche démarrée du serveur mqweb a besoin de:

- Identificateur utilisateur z/OS UNIX (UID) permettant d'utiliser z/OS UNIX System Services.
- Accès aux fichiers `hlq.SCSQAUTH` et `hlq.SCSQANL*` dans l'installation IBM MQ .
- Accès en lecture aux fichiers d'installation IBM MQ dans z/OS UNIX System Services.
- Accès en lecture et en écriture au répertoire utilisateur Liberty créé par le script **`crtmqweb`** .
- Droit de connexion au gestionnaire de files d'attente. Accordez à l'ID utilisateur de la tâche démarrée du serveur mqweb l'accès *READ* au profil `hlq.BATCH` dans la classe MQCONN.
- Droit d'émettre des commandes IBM MQ et d'accéder à certaines files d'attente. Ces détails sont décrits dans «IBM MQ Console -profils de sécurité de commande requis», à la page 235, «Sécurité de la file d'attente système», à la page 209et «Profils pour la sécurité de contexte», à la page 221.
- Droit d'abonnement à la rubrique SYSTEM.FTE , afin d'utiliser REST API for MFT. Accordez à l'ID utilisateur de la tâche démarrée du serveur mqweb l'accès *ALTER* au profil `hlq.SUBSCRIBE.SYSTEM.FTE` dans la classe MXTOPIC.
- Si vous configurez un registre SAF, accédez à divers profils de sécurité. Pour plus d'informations, voir «Configuration d'un registre SAF pour IBM MQ Console et REST API», à la page 515.

Authentification de connexion

Si votre gestionnaire de files d'attente a été configuré pour exiger que toutes les applications par lots fournissent un ID utilisateur et un mot de passe valides, en définissant CHKLOCL (REQUIRED), vous devez accorder à l'ID utilisateur de la tâche démarrée du serveur mqweb l'accès *UPDATE* au profil `hlq.BATCH` dans la classe MQCONN.

Ce droit entraîne le fonctionnement de l'authentification de connexion en mode CHKLOCL (FACULTATIF) pour l'ID utilisateur de la tâche démarrée du serveur mqweb.

Si vous n'avez pas configuré le gestionnaire de files d'attente pour exiger que toutes les applications par lots fournissent un ID utilisateur et un mot de passe valides, il suffit de donner à l'ID utilisateur qui démarre la tâche du serveur mqweb l'accès *READ* au profil `hlq.BATCH` dans la classe MQCONN.

Pour plus d'informations sur CHCKLOCL, voir «Utilisation de CHCKLOCL sur des applications liées localement», à la page 199.

Accès aux ressources IBM MQ requises pour utiliser MQ Console ou REST API

Les opérations effectuées dans MQ Consoleou REST APIpar un utilisateur ayant le rôle `MQWebUser` sont effectuées dans le contexte de sécurité de l'utilisateur.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur les rôles dans MQ Console et REST API, voir «Rôles sur les IBM MQ Console et REST API», à la page 518 .

Utilisez la procédure suivante pour accorder à un utilisateur, dans le rôle `MQWebUser` , l'accès aux ressources de gestionnaire de files d'attente requises pour utiliser MQ Console ou REST API.

Procédure

1. Accordez à l'ID utilisateur `mqweb server started task` un autre accès utilisateur à chaque ID utilisateur du rôle `MQWebUser` .

   Effectuez cette opération sur chaque gestionnaire de files d'attente que les utilisateurs administreront via MQ Console ou REST API.

   Vous pouvez utiliser les exemples de commande RACF suivants pour accorder à l'ID utilisateur `mqweb server started task` un accès utilisateur de remplacement à un utilisateur ayant le rôle `MQWebUser` :

   ```
   RDEFINE MQADMIN hlq.ALTERNATE.USER.userId UACC(NONE)
   PERMIT hlq.ALTERNATE.USER.userId CLASS(MQADMIN) ACCESS(UPDATE) ID(mqwebUserId)
   SETROPTS RACLIST(MQADMIN) REFRESH
   ```

   où :

   `hlq`
   Correspond au préfixe de profil, qui peut être le nom du gestionnaire de files d'attente ou le nom du groupe de partage de files d'attente

   `userId`
   Est l'utilisateur ayant le rôle `MQWebUser`

   `mqwebUserId`
   Est l'ID utilisateur `mqweb server started task`

   Remarque : Si vous utilisez la sécurité à casse mixte, utilisez la classe MXADMIN plutôt que la classe MQADMIN.
2. Accordez à chaque utilisateur du rôle `MQWebUser` l'accès aux files d'attente système nécessaires pour utiliser MQ Console et REST API.

Pour ce faire, pour les deux systèmes SYSTEM.ADMIN.COMMAND.QUEUE et SYSTEM.REST.REPLY.QUEUE, accordez à chaque utilisateur l'accès UPDATE aux classes MQQUEUE ou MXQUEUE, selon que la sécurité à casse mixte est utilisée ou non.

Vous devez effectuer cette opération sur chaque gestionnaire de files d'attente que l'utilisateur administrera via REST API, y compris les gestionnaires de files d'attente éloignées gérés via la passerelle administrative REST API.

3. Pour permettre à un utilisateur ayant le rôle `MQWebUser` d'administrer des gestionnaires de files d'attente éloignées, accordez à l'utilisateur l'accès UPDATE au profil dans la classe MQQUEUE ou MXQUEUE, en protégeant la file d'attente de transmission utilisée pour envoyer des commandes au gestionnaire de files d'attente éloignées. Notez que vous devez accorder à l'utilisateur l'accès UPDATE sur le gestionnaire de files d'attente de passerelle.

   Sur le gestionnaire de files d'attente éloignées, accordez l'accès au même utilisateur pour l'insertion dans la file d'attente de transmission utilisée pour renvoyer les messages de réponse de commande au gestionnaire de files d'attente de passerelle.
4. Accordez aux utilisateurs du rôle `MQWebUser` l'accès à toutes les autres ressources requises pour effectuer les opérations prises en charge par MQ Console et REST API.

   L'accès nécessaire pour:

   - L'exécution d'opérations dans le REST APIest décrite dans les sections *Exigences de sécurité* des ressources REST API individuelles
   - L'exécution de commandes par le MQ Console est décrite dans «IBM MQ Console -profils de sécurité de commande requis», à la page 235

V 9.1.0 Configuration de TLS pour REST API et IBM MQ Console sous z/OS

Sous z/OS, vous pouvez configurer le serveur mqweb pour qu'il utilise un fichier de clés RACF afin de stocker des certificats pour des connexions sécurisées avec TLS et l'authentification par certificat client.

Avant de commencer

Vous devez être un utilisateur disposant d'un accès en écriture au fichier `mqwebuser.xml` et d'un droit d'utilisation des fichiers de clés SAF pour exécuter cette procédure.

Pourquoi et quand exécuter cette tâche

La configuration du serveur mqweb par défaut utilise les magasins de clés Java pour le serveur et les certificats de confiance. Sous z/OS, vous pouvez configurer le serveur mqweb pour qu'il utilise un fichier de clés RACF à la place des magasins de clés Java . Le serveur peut également être configuré pour permettre aux utilisateurs de s'authentifier à l'aide d'un certificat client.

Pour plus d'informations sur l'utilisation des fichiers de clés RACF dans Liberty, voir Liberty: Magasins de clés .

Suivez cette procédure pour configurer le serveur mqweb afin qu'il utilise un fichier de clés RACF et, le cas échéant, pour configurer l'authentification par certificat client.

Procédure

1. Créez un certificat d'autorité de certification qui sera utilisé pour signer le certificat du serveur. Par exemple, entrez la commande RACF suivante:

```
RACDCERT GENCERT
  CERTAUTH
  SUBJECTSDN(CN('mqweb Certification Authority')
    O('IBM')
    OU('MQ'))
  SIZE(2048)
  WITHLABEL('mqwebCertauth')
```

2. Créez un certificat serveur, signé avec le certificat de l'autorité de certification créé à l'étape 1, en entrant la commande suivante:

```
RACDCERT ID(mqwebUserId) GENCERT
  SUBJECTSDN(CN('hostname')
   O('IBM')
   OU('MQ'))
   SIZE(2048)
   SIGNWITH (CERTAUTH LABEL('mqwebCertauth'))
  WITHLABEL('mqwebServerCert')
```

   où *mqwebUserId* est l'ID utilisateur de la tâche démarrée du serveur mqweb et *nom_hôte* est le nom d'hôte du serveur mqweb.
3. Connectez le certificat de l'autorité de certification et le certificat serveur à un fichier de clés SAF en entrant les commandes suivantes:

```
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebCertauth') CERTAUTH)
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebServerCert'))
```

   où *mqwebUserId* est l'ID utilisateur de la tâche démarrée du serveur mqweb et *keyring* est le nom du fichier de clés que vous souhaitez utiliser.
4. Exportez le certificat de l'autorité de certification dans un fichier CER en entrant la commande suivante:

```
RACDCERT CERTAUTH EXPORT(LABEL('mqwebCertauth'))
        DSN('hlq.CERT.MQWEBCA')
        FORMAT(CERTDER)
        PASSWORD('password')
```

5. Envoyez par FTP le certificat de l'autorité de certification exporté en binaire sur votre poste de travail et importez-le dans votre navigateur en tant que certificat de l'autorité de certification.
6. Facultatif : Si vous souhaitez configurer l'authentification par certificat client, créez et exportez un certificat client.
   a) Créez un certificat d'autorité de certification qui sera utilisé pour signer le certificat client. Par exemple, entrez la commande RACF suivante:

```
RACDCERT GENCERT
  CERTAUTH
  SUBJECTSDN(CN('mqweb User CA')
   O('IBM')
   OU('MQ'))
   SIZE(2048)
  WITHLABEL('mqwebUserCertauth')
```

   b) Connectez le certificat de l'autorité de certification à un fichier de clés SAF en entrant la commande suivante:

```
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebUserCertauth') CERTAUTH)
```

      où *mqwebUserId* est l'ID utilisateur de la tâche démarrée du serveur mqweb et *keyring* est le nom du fichier de clés que vous souhaitez utiliser.
   c) Créez un certificat client signé avec le certificat de l'autorité de certification. Par exemple, entrez la commande suivante :

```
RACDCERT ID(clientUserId) GENCERT
  SUBJECTSDN(CN('clientUserId')
     O('IBM')
     OU('MQ'))
   SIZE(2048)
   SIGNWITH (CERTAUTH LABEL('mqwebUserCertauth'))
  WITHLABEL('userCertLabel')
```

où *clientUserId* est le nom d'utilisateur.

La méthode utilisée pour mapper un certificat à un principal dépend du type de registre d'utilisateurs configuré:

- Si vous utilisez un registre de base, la zone Nom usuel du certificat est mise en correspondance avec l'utilisateur du registre.
- Si vous utilisez un registre SAF et que le certificat se trouve dans la base de données RACF , le propriétaire du certificat, spécifié avec le paramètre **ID** lors de la création du certificat, est utilisé.
- Si vous utilisez un registre LDAP, le nom distinctif complet du certificat est comparé au registre LDAP.

d) Exportez le certificat client dans un fichier PKCS #12 en entrant la commande suivante:

```
RACDCERT ID(mqwebUserId) EXPORT(LABEL('userCertLabel')) PASSWORD('password')
DSN('hlq.USER.CERT')
```

e) Envoyez par FTP le certificat exporté en binaire sur votre poste de travail. Pour utiliser le certificat client avec le IBM MQ Console, importez-le dans le navigateur Web utilisé pour accéder au IBM MQ Console en tant que certificat personnel.

7. Editez le fichier *WLP_user_directory*/servers/mqweb/mqwebuser.xml, où *WLP_user_directory* est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb** pour créer la définition de serveur mqweb.

   Effectuez les modifications suivantes pour configurer le serveur mqweb afin qu'il utilise un fichier de clés RACF :

   a) Supprimez ou mettez en commentaire la ligne suivante:

   ```
   <sslDefault sslRef="mqDefaultSSLConfig"/>
   ```

   b) Ajoutez les instructions suivantes:

   ```
   <keyStore id="defaultKeyStore" filebased="false" location="safkeyring://mqwebUserId/
   keyring"
             password="password" readOnly="true" type="JCERACFKS" />
   <ssl id="thisSSLConfig" keyStoreRef="defaultKeyStore" sslProtocol="TLSv1.2"
             serverKeyAlias="mqwebServerCert" clientAuthenticationSupported="true" />
   <sslDefault sslRef="thisSSLConfig"/>
   ```

   où :

   - *mqwebUserID* est l'ID utilisateur de la tâche démarrée du serveur mqweb.
   - *keyring* est le nom du fichier de clés RACF .
   - *mqwebServerCert* est le libellé du certificat du serveur mqweb.

   Remarques : La valeur de **keyStore password** est ignorée.

8. Redémarrez le serveur mqweb en arrêtant et en redémarrant la tâche démarrée du serveur mqweb.
9. Facultatif : Utilisez le certificat client pour l'authentification:

   - Pour utiliser le certificat client avec le IBM MQ Console, entrez l'URL du MQ Console dans le navigateur Web où vous avez installé le certificat client.
   - Pour utiliser le certificat client avec l'API REST, indiquez le certificat client avec chaque demande REST.

   Remarques :

   a. Si vous utilisez uniquement des certificats pour vous authentifier auprès du IBM MQ Console, le navigateur peut afficher une liste de certificats que vous pouvez sélectionner.

   b. Si vous souhaitez utiliser un autre certificat, vous devrez peut-être fermer et redémarrer votre navigateur.

c. Si vous utilisez des certificats client qui ne se trouvent pas dans la base de données RACF , vous pouvez utiliser le filtrage des noms de certificat RACF pour mapper des attributs de certificat à un ID utilisateur. Exemple :

```
RACDCERT ID(DEPT3USR) MAP SDNFILTER(OU=DEPT1.C=US)
```

mappe les certificats avec un nom distinctif de sujet contenant OU=DEPT1 et C=US à l'ID utilisateur DEPT3USR.

Résultats

Vous avez configuré une interface TLS pour IBM MQ Console et REST API.

ULW Gestion des clés et des certificats sur UNIX, Linux, and Windows

Utilisez la commande `runmqckm` (UNIX et Windows) et la commande `runmqakm` (UNIX, Linux, and Windows) pour gérer les clés, les certificats et les demandes de certificat.

Commande runmqckm

La commande **runmqckm** est disponible sous UNIX et Windows.

La commande **runmqckm** fournit des fonctions similaires à celles de iKeyman, décrites dans «Sécurisation de IBM MQ», à la page 5.

Pour utiliser la commande **runmqckm** , vérifiez que les variables d'environnement système sont correctement configurées en exécutant la commande **setmqenv** .

V 9.1.0 La commande **runmqckm** requiert l'installation du composant IBM MQ JRE. Si ce composant n'est pas installé, vous pouvez utiliser la commande **runmqackm** à la place.

Commande runmqakm

La commande **runmqakm** est disponible sous UNIX, Linuxet Windows.

Pour utiliser la commande **runmqaqm** , vérifiez que les variables d'environnement système sont correctement configurées en exécutant la commande **setmqenv** .

Si vous devez gérer les certificats TLS conformément à la norme FIPS, utilisez la commande **runmqakm** à la place des commandes **runmqckm** . En effet, la commande **runmqakm** prend en charge un chiffrement renforcé.

Utilisez les commandes `runmqckm` et `runmqakm` pour effectuer les opérations suivantes:

- Créez le type de fichiers de base de données de clés CMS requis par IBM MQ
- Créer des demandes de certificat
- Importer des certificats personnels
- Importer des certificats d'autorité de certification
- Gérer les certificats autosignés

Information associée

Keytool

ULW runmqckm et commandes runmqakm sous UNIX, Linux, and Windows

Cette section décrit les commandes runmqckm et runmqakm en fonction de l'objet de la commande.

Les principales différences entre les deux commandes sont les suivantes:

- ULW **runmqakm**

  - Est disponible sous UNIX, Linuxet Windows.
  - Prend en charge la création de certificats et de demandes de certificat avec des clés publiques Elliptic Curve, contrairement à la commande **runmqckm** .
  - Prend en charge un chiffrement plus fort du fichier de référentiel de clés que la commande **runmqckm** via le paramètre **-strong** .
  - A été certifié conforme à la norme FIPS 140-2, et peut être configuré pour fonctionner de manière conforme à la norme FIPS, à l'aide du paramètre **-fips** , contrairement à la commande **runmqckm** .
- Windows UNIX **runmqckm**
  - Est disponible sous UNIX et Windows.
  - Prend en charge les formats de fichier de référentiel de clés JKS et JCEKS, contrairement à la commande **runmqakm** .

Avertissement : V 9.1.0 La commande **runmqckm** requiert l'installation de la fonction IBM MQ Java runtime environment (JRE).

Chaque commande spécifie au moins un *objet*. Les commandes pour les opérations d'unité PKCS #11 peuvent spécifier des objets supplémentaires. Les commandes des objets de base de données de clés, de certificat et de demande de certificat spécifient également une *action*. L'objet peut être l'un des suivants:

-keydb
Les actions s'appliquent à une base de données de clés

-cert
Les actions s'appliquent à un certificat

-certreq
Les actions s'appliquent à une demande de certificat

-aide
Affiche l'aide

-version
Affiche les informations de version

Les sous-rubriques suivantes décrivent les actions que vous pouvez effectuer sur les objets de base de données de clés, de certificat et de demande de certificat. Voir «Options runmqckm et runmqakm sous UNIX, Linux, and Windows», à la page 545 pour une description des options de ces commandes.

ULW Commandes pour une base de données de clés CMS uniquement sous UNIX, Linux, and Windows

Vous pouvez utiliser les commandes **runmqckm**et **runmqakm** pour gérer les clés et les certificats d'une base de données de clés CMS.

-keydb -changepw
Modifiez le mot de passe d'une base de données de clés CMS:

```
-keydb -changepw -db filename -pw password -new_pw new_password


-stash
```

-keydb -créer
Créez une base de données de clés CMS:

```
-keydb -create -db filename
-pw password -type cms -expire days -stash
```

-keydb -stashpw
Stockez le mot de passe d'une base de données de clés CMS dans un fichier:

```
-keydb -stashpw -db filename
-pw password
```

-cert -getdefault

Remarque : Le certificat par défaut n'est pas pris en charge par IBM MQ 8.0. Vous devez utiliser la configuration de libellé de certificat comme décrit dans «Labels de certificat numérique, compréhension des exigences», à la page 26.

Obtenez le certificat personnel par défaut:

```
-cert -getdefault -db filename
-pw password
```

-cert-modifier

Modifier un certificat.

Remarque : Actuellement, la seule zone pouvant être modifiée est la zone Certificate Trust.

```
-cert -modify -db filename
-pw password -label label
-trust enable|disable
```

-cert -setdefault

Remarque : Le certificat par défaut n'est pas pris en charge par IBM MQ 8.0 ou version ultérieure. Vous devez utiliser la configuration de libellé de certificat comme décrit dans «Labels de certificat numérique, compréhension des exigences», à la page 26.

Définissez le certificat personnel par défaut:

```
-cert -setdefault -db filename
-pw password -label label
```

ULW Commande pour les bases de données de clés CMS ou PKCS #12 sous UNIX, Linux, and Windows

Vous pouvez utiliser les commandes runmqckm et runmqakm pour gérer les clés et les certificats d'une base de données de clés CMS ou PKCS #12 .

Remarque : IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .

Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.

-keydb -changepw

Modifiez le mot de passe d'une base de données de clés:

```
-keydb -changepw -db filename -pw password -new_pw
new_password -expire days
```

-keydb -convert

convertissez la base de données de clés d'un format à un autre:

```
-keydb -convert -db filename -pw password
-old_format cms | pkcs12 -new_format cms
```

-keydb -créer

Créez une base de données de clés:

```
-keydb -create -db filename -pw password -type cms
| pkcs12
```

-keydb -delete
Supprimez une base de données de clés:

```
-keydb -delete -db filename -pw password
```

-keydb -liste
Répertorier les types de base de données de clés actuellement pris en charge:

```
-keydb -list
```

-cert -add
Ajoutez un certificat d'un fichier dans une base de données de clés:

```
-cert -add -db filename -pw password -label label
-file filename
-format ascii | binary
```

-cert -create
Créez un certificat autosigné:

```
-cert -create -db filename -pw password -label label
-dn distinguished_name
-size 1024 | 512 -x509version 3 | 1
| 2
-expire days -sig_alg MD2_WITH_RSA | MD2WithRSA
|
MD5_WITH_RSA | MD5WithRSA
|
SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA
|
SHA2WithRSA | SHA384_WITH_RSA
|
SHA384WithRSA | SHA512_WITH_RSA
|
SHA512WithRSA | SHA_WITH_DSA
|
SHA_WITH_RSA | SHAWithDSA
|
SHAWithRSA
```

-cert -delete
Supprimer un certificat:

```
-cert -delete -db filename -pw password -label label
```

-cert -details
Répertoriez les informations détaillées pour un certificat spécifique:

```
-cert -details -db filename -pw password -label label
```

-cert -export
Exportez un certificat personnel et sa clé privée associée à partir d'une base de données de clés dans un fichier PKCS #12 ou dans une autre base de données de clés:

```
-cert -export -db filename -pw password -label label
-type cms | pkcs12
```

```
-target filename -target_pw password -target_type
cms | pkcs12
```

-cert -extract

Extrayez un certificat d'une base de données de clés:

```
-cert -extract -db filename -pw password -label label
-target filename
-format ascii | binary
```

-cert -import

Importez un certificat personnel à partir d'une base de données de clés:

```
-cert -import -file filename -pw password -type
pkcs12 -target filename
-target_pw password -target_type cms -label
label
```

L'option `-label` est obligatoire et spécifie le libellé du certificat à importer de la base de données de clés source.

L'option `-new_label` est facultative et permet d'attribuer au certificat importé un libellé différent dans la base de données de clés cible de celui de la base de données source.

-cert -list

Répertoriez tous les certificats d'une base de données de clés:

```
-cert -list all | personal | CA
-db filename -pw password
```

-cert -receive

Recevoir un certificat d'un fichier:

```
-cert -receive -file filename -db filename -pw password

-format ascii | binary -default_cert yes |
no
```

-cert -signe

Signer un certificat:

```
-cert -sign -db filename -file filename -pw password
-label label -target filename
-format ascii | binary -expire days
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

-certreq -create

Créez une demande de certificat:

```
-certreq -create -db filename -pw password
-label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA |
MD5_WITH_RSA | MD5WithRSA |
SHA1WithDSA | SHA1WithRSA |
SHA256_WITH_RSA | SHA256WithRSA |
```

```
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

-certreq -delete
Supprimez une demande de certificat:

```
-certreq -delete -db filename -pw password -label
label
```

-certreq -details
Répertoriez les informations détaillées d'une demande de certificat spécifique:

```
-certreq -details -db filename -pw password -label
label
```

Répertoriez les informations détaillées sur une demande de certificat et affichez la demande de certificat complète:

```
-certreq -details -showOID -db filename
-pw password -label label
```

-certreq -extract
Extrayez une demande de certificat d'une base de données de demandes de certificat dans un fichier:

```
-certreq -extract -db filename -pw password
-label label -target filename
```

-certreq -list
Répertoriez toutes les demandes de certificat dans la base de données des demandes de certificat:

```
-certreq -list -db filename -pw password
```

-certreq -recréer
Recréez une demande de certificat:

```
-certreq -recreate -db filename -pw password
-label label -target filename
```

ULW Commandes pour les opérations d'unité de chiffrement sous UNIX, Linux, and Windows

Vous pouvez utiliser les commandes runmqckm et runmqakm pour gérer les clés et les certificats pour les opérations d'unité de chiffrement.

Remarque : IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .

Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.

-keydb -changepw
Modifiez le mot de passe d'une unité de chiffrement:

```
-keydb -changepw -crypto module_name -tokenlabel token_label
-pw password -new_pw new_password
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-keydb -liste

Répertorier les types de base de données de clés actuellement pris en charge:

```
-keydb -list
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -add

Ajoutez un certificat d'un fichier à une unité de chiffrement:

```
-cert -add -crypto module_name -tokenlabel token_label
-pw password -label label -file filename -format
ascii | binary
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -create

Créez un certificat autosigné sur une unité de chiffrement:

```
-cert -create -crypto module_name -tokenlabel token_label

-pw password -label label -dn distinguished_name
-size 1024 | 512
-x509version 3 | 1 | 2 -default_cert no
| yes -expire days
-sig_alg MD2_WITH_RSA | MD2WithRSA |
MD5_WITH_RSA | MD5WithRSA |
SHA1WithDSA | SHA1WithRSA |
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

Remarque : Vous ne pouvez pas importer un certificat contenant plusieurs attributs d'unité organisationnelle (unité organisationnelle) dans le nom distinctif.

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -delete

Supprimez un certificat sur une unité de chiffrement:

```
-cert -delete -crypto module_name -tokenlabel token_label
-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -details

Répertoriez les informations détaillées relatives à un certificat spécifique sur une unité de chiffrement:

```
-cert -details -crypto module_name -tokenlabel token_label

-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

Répertoriez les informations détaillées et affichez le certificat complet d'un certificat spécifique sur une unité de chiffrement:

```
-cert -details -showOID -crypto module_name -tokenlabel
token_label
-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -extract

Extrayez un certificat d'une base de données de clés:

```
-cert -extract -crypto module_name -tokenlabel token_label
-pw password -label label -target filename
-format ascii | binary
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -import

Importez un certificat sur une unité de chiffrement avec prise en charge de la base de données de clés secondaires:

```
-cert -import -db filename -pw password -label label
-type cms
-crypto module_name -tokenlabel token_label -pw
```

```
password
-secondaryDB filename -secondaryDBpw password
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

```
-cert -import -db filename -pw password -label label
-type cms
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password -fips
```

Importez un certificat PKCS #12 sur une unité de chiffrement avec prise en charge de la base de données de clés secondaires:

```
-cert -import -file filename -pw password -type pkcs12
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

```
-cert -import -file filename -pw password -type pkcs12
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password -fips
```

Remarque : Vous ne pouvez pas importer un certificat contenant plusieurs attributs d'unité organisationnelle (unité organisationnelle) dans le nom distinctif.

-cert -list

Répertorier tous les certificats sur une unité de chiffrement:

```
-cert -list all | personal | CA
-crypto module_name -tokenlabel token_label -pw
password
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-cert -receive

Réception d'un certificat d'un fichier vers une unité de chiffrement avec prise en charge de la base de données de clés secondaires:

```
-cert -receive -file filename -crypto module_name -tokenlabel
token_label
-pw password -default_cert yes | no
```

```
-secondaryDB filename -secondaryDBpw password -format
ascii | binary
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

A l'aide de la commande **runmqakm** :

-certreq -create
Créez une demande de certificat sur une unité de chiffrement:

```
-certreq -create -crypto module_name -tokenlabel token_label

-pw password -label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

Remarque : Vous ne pouvez pas importer un certificat contenant plusieurs attributs d'unité organisationnelle (unité organisationnelle) dans le nom distinctif.

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-certreq -delete
Supprimez une demande de certificat d'une unité de chiffrement:

```
-certreq -delete -crypto module_name -tokenlabel token_label

-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-certreq -details
Répertoriez les informations détaillées d'une demande de certificat spécifique sur une unité de chiffrement:

```
-certreq -details -crypto module_name -tokenlabel token_label

-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une

bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

Répertoriez les informations détaillées sur une demande de certificat et affichez la demande de certificat complet sur une unité de chiffrement:

```
-certreq -details -showOID -crypto module_name -tokenlabel
token_label
-pw password -label label
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-certreq -extract
Extrayez une demande de certificat d'une base de données de demandes de certificat sur une unité de chiffrement dans un fichier:

```
-certreq -extract -crypto module_name -tokenlabel token_label

-pw password -label label -target filename
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

-certreq -list
Répertorier toutes les demandes de certificat dans la base de données des demandes de certificat sur une unité de chiffrement:

```
-certreq -list -crypto module_name -tokenlabel token_label

-pw password
```

Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11, notez que **runmqckm** et **strmqikm** sont des programmes 64 bits. Etant donné que les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans un processus 64 bits, une bibliothèque PKCS #11 64 bits doit être installée pour l'administration du matériel de cryptographie. Les plateformes Windows et Linux x86 32 bits sont les seules exceptions, car les programmes **strmqikm** et **runmqckm** sont des programmes 32 bits sur ces plateformes.

ULW Options runmqckm et runmqakm sous UNIX, Linux, and Windows

Vous pouvez utiliser les options de ligne de commande **runmqckm** (iKeycmd) et **runmqakm** pour gérer les clés, les certificats et les demandes de certificat.

ULW La commande `runmqakm` est disponible sur UNIX, Linux, and Windows.

Windows UNIX La commande `runmqckm` est disponible sous UNIX et Windows.

Remarque : IBM MQ ne prend pas en charge les algorithmes SHA-3 ou SHA-5 . Vous pouvez utiliser les noms d'algorithme de signature numérique SHA384WithRSA et SHA512WithRSA car ces deux algorithmes sont membres de la famille SHA-2 .

Les noms d'algorithme de signature numérique SHA3WithRSA et SHA5WithRSA sont obsolètes car ils sont de forme abrégée SHA384WithRSA et SHA512WithRSA respectivement.

La signification d'une option peut dépendre de l'objet et de l'action spécifiés dans la commande.

*Tableau 90. Options pouvant être utilisées avec **runmqckm** et **runmqakm***

Paramètre	Description
`-create`	Option permettant de créer une base de données de clés.
`-crypto`	Nom du module permettant de gérer une unité de chiffrement PKCS #11 . La valeur après **-crypto** est facultative si vous spécifiez le nom du module dans le fichier de propriétés. Si vous utilisez des certificats ou des clés stockés sur du matériel de cryptographie PKCS #11 , notez que **runmqckm** et **strmqikm** sont exécutés à l'aide de la machine virtuelle Java (JVM) fournie avec l'installation IBM MQ . Les modules externes requis pour la prise en charge de PKCS #11 seront chargés dans le processus JVM. Par conséquent, vous devez disposer d'une bibliothèque PKCS #11 installée pour l'administration du matériel de cryptographie qui correspond au nombre de bits de la machine JVM et spécifier cette bibliothèque dans **runmqckm** ou **strmqikm**.
-db	Nom de chemin qualifié complet d'une base de données de clés.
-default_cert	Définit un certificat comme certificat par défaut. La valeur peut être `yes` ou `no`. La valeur par défaut est `NO`.
-dn	Nom distinctif X.500 . La valeur est une chaîne entre guillemets, par exemple `"CN=John Smith,O=IBM,OU=Test,C=GB"`. Notez que seuls les attributs O et C sont requis. La spécification d'un nom usuel (CN) est facultative.
-encryption	Niveau de chiffrement utilisé dans la commande d'exportation de certificat. La valeur peut être `forte` ou `faible`. La valeur par défaut est `strong`.
-expire	Délai d'expiration, en jours, d'un certificat ou d'un mot de passe de base de données. La valeur par défaut est 365 jours pour un mot de passe de certificat. Il n'y a pas de temps par défaut pour un mot de passe de base de données: utilisez le paramètre **-expire** pour définir explicitement un délai d'expiration de mot de passe de base de données.
-file	Nom de fichier d'un certificat ou d'une demande de certificat.
-fips	indique que la commande est exécutée en mode FIPS. En mode FIPS, le composant ICC utilise des algorithmes validés par FIPS 140-2. Si le composant ICC ne s'initialise pas en mode FIPS, la commande **runmqakm** échoue.
-format	Format d'un certificat. La valeur peut être `ascii` pour Base64_encoded ASCII ou `binary` pour les données DER binaires. La valeur par défaut est `ascii`.
-label	Libellé associé à un certificat ou à une demande de certificat. Si le certificat est un certificat personnel utilisé pour identifier une application client ou un gestionnaire de files d'attente IBM MQ , le libellé doit correspondre au paramètre de libellé de certificat IBM MQ (CERTLABL). Pour plus d'informations, voir «Labels de certificat numérique, compréhension des exigences», à la page 26.
-new_format	Nouveau format de la base de données de clés.

*Tableau 90. Options pouvant être utilisées avec **runmqckm** et **runmqakm** (suite)*	
Paramètre	**Description**
-new_label	Utilisée dans une commande d'importation de certificat, cette option permet d'importer un certificat avec un libellé différent de celui qu'il avait dans la base de données de clés source. Si le certificat est un certificat personnel utilisé pour identifier une application client ou un gestionnaire de files d'attente IBM MQ , le libellé doit correspondre au paramètre de libellé de certificat IBM MQ (CERTLABL). Pour plus d'informations, voir «Labels de certificat numérique, compréhension des exigences», à la page 26.
-new_pw	Nouveau mot de passe de base de données.
-old_format	Ancien format de la base de données de clés.
-pw	Mot de passe de la base de données de clés ou du fichier PKCS #12 .
-secondaryDB	Nom d'une base de données de clés secondaire pour les opérations sur les unités PKCS #11 .
-secondaryDBpw	Mot de passe de la base de données de clés secondaire pour les opérations sur les unités PKCS #11 .
-showOID	Affiche le certificat complet ou la demande de certificat.
-sig_alg	Algorithme de hachage utilisé lors de la création d'une demande de certificat, d'un certificat autosigné ou de la signature d'un certificat. Cet algorithme de hachage est utilisé pour créer la signature associée au nouveau certificat ou à la nouvelle demande de certificat. Pour **runmqckm**, la valeur peut être MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. La valeur par défaut est SHA1WithRSA. Pour **runmqakm**, la valeur peut être md5, MD5_WITH_RSA, MD5WithRSA, SHA_WITH_DSA, SHA_WITH_RSA, sha1, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, sha224, SHA224_WITH_RSA, SHA224WithDSA, SHA224WithECDSA, SHA224WithRSA, sha256, SHA256_WITH_RSA, SHA256WithDSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithRSA, sha384, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, sha512, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHAWithDSA, SHAWithRSA, EC_ecdsa_with_SHA1, EC_ecdsa_with_SHA224, EC_ecdsa_with_SHA256, EC_ecdsa_with_SHA384ou EC_ecdsa_with_SHA512. La valeur par défaut est SHA1WithRSA.

*Tableau 90. Options pouvant être utilisées avec **runmqckm** et **runmqakm** (suite)*	
Paramètre	**Description**
-size	Taille de la clé. Pour **runmqckm**, la valeur peut être 512, 1024 ou 2048. La valeur par défaut est 1024 bits. Pour **runmqakm**, la valeur dépend de l'algorithme de signature: • Pour les algorithmes de signature RSA (l'algorithme par défaut utilisé si aucun **-sig_alg** n'est spécifié), la valeur peut être 512, 1024, 2048 ou 4096. Une taille de clé RSA de 512 bits n'est pas autorisée si le paramètre **-fips** est activé. La taille de clé RSA par défaut est de 1024 bits. • Pour les algorithmes de courbe elliptique, la valeur peut être 256, 384 ou 512. La taille de clé Elliptic Curve par défaut dépend de l'algorithme de signature. Pour SHA256, il s'agit de 256 ; pour SHA384, il s'agit de 384 ; pour SHA512, il s'agit de 512.
-stash	Stockez le mot de passe de la base de données de clés dans un fichier. Applicable uniquement aux bases de données de type CMS et PKCS12. **Remarque : -stash** est valide sur les commandes **-keydb** -create pour indiquer à **runmqckm**/**runmqakm** de créer un fichier de dissimulation contenant le mot de passe. L'exécution de la commande `$ runmqakm -help` répertorie uniquement les paramètres d'aide de haut niveau.
-stashed	Indique que le mot de passe de la base de données de clés ou du fichier PKCS #12 se trouve dans un fichier de dissimulation. **Remarque :** L'option **-stashed** est valide pour les appels autres que les commandes **-keydb** -create. Si vous ne spécifiez pas cette option, vous devez indiquer le mot de passe à l'aide de **-pw**. En outre, ce n'est que lorsque vous indiquez à la commande le type d'action que vous effectuez que l'aide détaillée indiquant **-stashed** s'affiche.
-target	Fichier de destination ou base de données.
-target_pw	Mot de passe de la base de données de clés si **-target** spécifie une base de données de clés.
-target_type	Type de base de données spécifié par l'opérande **-target** . Pour connaître les valeurs admises, voir le paramètre **-type** .
-tokenLabel	Libellé d'une unité de chiffrement PKCS #11 .
-trust	Statut de confiance d'un certificat de l'autorité de certification. La valeur peut être `enable` ou `disable`. La valeur par défaut est `enable`.
-type	Type de base de données. La valeur peut être l'une des valeurs suivantes : • `cms` pour une base de données de clés CMS • `pkcs12` pour un fichier PKCS #12 .
-x509version	Version du certificat X.509 à créer. La valeur peut être 1, 2 ou 3. La valeur par défaut est 3.

*Tableau 90. Options pouvant être utilisées avec **runmqckm** et **runmqakm** (suite)*	
Paramètre	**Description**

-rfc3339	Utilisez ce paramètre pour générer la date au format RFC 3339 pour la commande `runmqakm -cert -details` , qui est au format suivant:
	`Not Before : 2015-08-26T08:53:37Z`
	`Not After : 2016-08-26T08:53:37Z`
	Notez que le paramètre **-rfc3339** doit apparaître dans la commande après les paramètres supplémentaires:
	`runmqakm -cert -details -db exampleDB -stashed -label certficateLabel -rfc3339`

Remarque : Les propriétés fournies avec le paramètre IBM Global Security Kit (GSKit) relatif au chiffrement par clé symétrique **-seckey** dans l'utilitaire **runmqckm** sont ignorées et ne sont pas prises en charge par IBM MQ.

ULW codes d'erreur runmqakm sous UNIX, Linux, and Windows

Tableau répertoriant les codes d'erreur numériques émis par runmqakm et leur signification.

Code d'erreur	Message d'erreur
0	Réussite
1	Une erreur inconnue s'est produite
2	Une erreur de codage / décodage ASN.1 s'est produite.
3	Une erreur s'est produite lors de l'initialisation du codeur / décodeur ASN.1 .
4	Une erreur de codage / décodage ASN.1 s'est produite en raison d'un index hors plage ou d'une zone facultative inexistante.
5	Une erreur de base de données s'est produite.
6	Une erreur s'est produite lors de l'ouverture du fichier de base de données, vérifiez l'existence du fichier et les droits d'accès.
7	Une erreur s'est produite lors de la réouverture du fichier de base de données.
8	Echec de la création de la base de données.
9	La base de données existe déjà.
10	Une erreur s'est produite lors de la suppression du fichier de base de données.
11	La base de données n'a pas pu être ouverte.
12	Une erreur s'est produite lors de la lecture du fichier de base de données.
13	Une erreur s'est produite lors de l'écriture des données dans le fichier base de données.

Code d'erreur	Message d'erreur
14	Une erreur de validation de la base de données s'est produite.
15	Une version de base de données non valide a été détectée.
16	Un mot de passe de base de données non valide a été détecté.
17	Un type de fichier de base de données non valide a été détecté.
18	La base de données spécifiée a été endommagée.
19	Un mot de passe non valide a été fourni ou la base de données de clés a été falsifiée ou endommagée.
20	Une erreur d'intégrité d'entrée de clé de base de données s'est produite.
21	Un certificat en double existe déjà dans la base de données.
22	Une clé en double existe déjà dans la base de données (ID enregistrement).
23	Un certificat portant le même libellé existe déjà dans la base de données de clés.
24	Une clé en double existe déjà dans la base de données (Signature).
25	Une clé en double existe déjà dans la base de données (certificat non signé).
26	Une clé en double existe déjà dans la base de données (émetteur et numéro de série).
27	Une clé en double existe déjà dans la base de données (informations sur la clé publique du sujet).
28	Une clé en double existe déjà dans la base de données (CRL non signée).
29	Le libellé a été utilisé dans la base de données.
30	Une erreur de chiffrement de mot de passe s'est produite.
31	Une erreur liée à LDAP s'est produite. (LDAP n'est pas pris en charge par ce programme)
32	Une erreur cryptographique s'est produite.
33	Une erreur de chiffrement / déchiffrement s'est produite.
34	Un algorithme de cryptographie non valide a été trouvé.
35	Une erreur s'est produite lors de la signature des données.
36	Une erreur s'est produite lors de la vérification des données.

Code d'erreur	Message d'erreur
37	Une erreur s'est produite lors du calcul du prétraitement des données.
38	Un paramètre cryptographique non valide a été trouvé.
39	Un algorithme de cryptographie non pris en charge a été détecté.
40	La taille d'entrée spécifiée est supérieure à la taille de module prise en charge.
41	Une taille de module non prise en charge a été trouvée.
42	Une erreur de validation de la base de données s'est produite.
43	La validation de l'entrée de clé a échoué.
44	Une zone d'extension en double existe.
45	La version de la clé est incorrecte.
46	Une zone d'extension obligatoire n'existe pas.
47	La période de validité ne comprend pas aujourd'hui ou ne tombe pas dans la période de validité de son émetteur
48	La période de validité n'inclut pas aujourd'hui ou ne tombe pas dans la période de validité de l'émetteur.
49	Une erreur s'est produite lors de la validation de l'extension d'utilisation de clé privée.
50	L'émetteur de la clé est introuvable.
51	Une extension de certificat requise est manquante.
52	Une extension de contrainte de base non valide a été trouvée.
53	La validation de la signature de la clé a échoué.
54	La clé racine de la clé n'est pas sécurisée.
55	La clé a été révoquée.
56	Une erreur s'est produite lors de la validation de l'extension de l'identificateur de clé de droits.
57	Une erreur s'est produite lors de la validation de l'extension d'utilisation de clé privée.
58	Une erreur s'est produite lors de la validation de l'extension de nom alternatif de sujet.
59	Une erreur s'est produite lors de la validation de l'extension de nom alternatif de l'émetteur.
60	Une erreur s'est produite lors de la validation de l'extension d'utilisation de clé.

Code d'erreur	Message d'erreur
61	Une extension critique inconnue a été trouvée.
62	Une erreur s'est produite lors de la validation des entrées de paire de clés.
63	Une erreur s'est produite lors de la validation de la CRL.
64	Une erreur d'exclusion mutuelle s'est produite.
65	Un paramètre non valide a été trouvé.
66	Une erreur d'allocation de mémoire ou de paramètre null a été détectée.
65	Le nombre ou la taille est trop grand ou trop petit.
68	L'ancien mot de passe n'est pas valide.
69	Le nouveau mot de passe n'est pas valide.
70	Le mot de passe a expiré.
71	Une erreur liée à l'unité d'exécution s'est produite.
84	Une erreur s'est produite lors de la création d'unités d'exécution.
73	Une erreur s'est produite alors qu'une unité d'exécution attendait de se fermer.
74	Une erreur d'entrée-sortie s'est produite.
75	Une erreur s'est produite lors du chargement de CMS.
76	Une erreur liée au matériel de cryptographie s'est produite.
77	La routine d'initialisation de la bibliothèque n'a pas été appelée.
78	La table de descripteur de base de données interne est endommagée.
79	Une erreur d'allocation de mémoire s'est produite.
80	Une option non reconnue a été trouvée.
81	Une erreur s'est produite lors de l'obtention des informations d'heure.
82	Une erreur de création de mutex s'est produite.
83	Une erreur s'est produite lors de l'ouverture du catalogue de messages.
84	Une erreur s'est produite lors de l'ouverture du catalogue de messages d'erreur
85	Un nom de fichier null a été trouvé.
86	Une erreur s'est produite lors de l'ouverture des fichiers, vérifiez l'existence du fichier et les droits d'accès.

Code d'erreur	Message d'erreur
87	Une erreur s'est produite lors de l'ouverture des fichiers à lire.
88	Une erreur s'est produite lors de l'ouverture des fichiers à écrire.
89	Il n'existe pas de fichier de ce type.
70	Le fichier ne peut pas être ouvert en raison de son paramètre de droits d'accès.
91	Une erreur s'est produite lors de l'écriture des données dans les fichiers.
92	Une erreur s'est produite lors de la suppression des fichiers.
93	Des données Base64-encoded non valides ont été trouvées.
94	Un type de message Base64 non valide a été trouvé.
95	Une erreur s'est produite lors du codage des données avec la règle de codage Base64 .
96	Une erreur s'est produite lors du décodage des données Base64-encoded .
97	Une erreur s'est produite lors de l'obtention d'une balise de nom distinctif.
98	La zone de nom usuel requise est vide.
99	La zone de nom de pays ou de région requise est vide.
100	Un descripteur de base de données non valide a été trouvé.
101	La base de données de clés n'existe pas.
102	La base de données de la paire de clés de demande n'existe pas.
103	Le fichier de mots de passe n'existe pas.
104	Le nouveau mot de passe est identique à l'ancien.
105	Aucune clé n'a été trouvée dans la base de données de clés.
106	Aucune clé de demande n'a été trouvée.
107	Aucune autorité de certification digne de confiance n'a été trouvée.
108	Aucune clé de demande n'a été trouvée pour le certificat.
109	Il n'y a pas de clé privée dans la base de données de clés.
110	Il n'y a pas de clé par défaut dans la base de données de clés.

Code d'erreur	Message d'erreur
111	Il n'y a pas de clé privée dans l'enregistrement de clé.
112	Il n'y a pas de certificat dans l'enregistrement de clé.
113	Il n'existe aucune entrée de liste de révocation de certificat.
114	Un nom de fichier de base de données de clés non valide a été trouvé.
115	Un type de clé privée non reconnu a été trouvé.
116	Une entrée de nom distinctif non valide a été trouvée.
117	Aucune entrée de clé ayant le libellé de clé spécifié n'a été trouvée.
118	La liste des libellés de clé a été endommagée.
119	Les données d'entrée ne sont pas des données PKCS12 valides.
120	Le mot de passe n'est pas valide ou les données PKCS12 ont été endommagées ou ont été créées avec une version ultérieure de PKCS12
121	Un type d'exportation de clé non reconnu a été trouvé.
122	Un algorithme de chiffrement par mot de passe non pris en charge a été trouvé.
123	Une erreur s'est produite lors de la conversion du fichier de clés en base de données de clés CMS.
124	Une erreur s'est produite lors de la conversion de la base de données de clés CMS en fichier de clés.
125	Une erreur s'est produite lors de la création d'un certificat pour la demande de certificat.
126	Une chaîne d'émetteur complète ne peut pas être générée.
127	Des données WEBDB non valides ont été trouvées.
128	Il n'y a aucune donnée à écrire dans le fichier de clés.
129	Le nombre de jours que vous avez entré s'étend au-delà de la période de validité autorisée.
130	Le mot de passe est trop court ; il doit comporter au moins {0} caractères.
131	Un mot de passe doit contenir au moins un chiffre.
132	Tous les caractères du mot de passe sont des caractères alphabétiques ou numériques.
133	Un algorithme de signature non reconnu ou non pris en charge a été spécifié.

Code d'erreur	Message d'erreur
134	Un type de base de données non valide a été détecté.
135 $	La base de données de clés secondaire spécifiée est utilisée par une autre unité PKCS#11 .
136	Aucune base de données de clés secondaires n'a été indiquée.
137	Le libellé n'existe pas sur l'unité PKCS#11 .
138	Mot de passe requis pour accéder à l'unité PKCS#11 .
139	Mot de passe non requis pour accéder à l'unité PKCS#11 .
140	Impossible de charger la bibliothèque cryptographique.
141	PKCS#11 n'est pas pris en charge pour cette opération.
142	Une opération sur une unité PKCS#11 a échoué.
143	L'utilisateur LDAP n'est pas un utilisateur valide. (LDAP n'est pas pris en charge par ce programme)
144	L'utilisateur LDAP n'est pas un utilisateur valide. (LDAP n'est pas pris en charge par ce programme)
145	La requête LDAP a échoué. (LDAP n'est pas pris en charge par ce programme)
146	Une chaîne de certificats non valide a été trouvée.
147	Le certificat racine n'est pas digne de confiance.
148	Un certificat révoqué a été détecté.
149	Une fonction d'objet cryptographique a échoué.
150	Aucune source de données de liste de révocation de certificat n'est disponible.
151	Aucun jeton cryptographique n'est disponible.
152	Le mode FIPS n'est pas disponible.
153	Il existe un conflit avec les paramètres du mode FIPS.
154	Le mot de passe entré ne correspond pas à la force minimale requise.
200	Un incident s'est produit lors de l'initialisation du programme.
201	Le marquage sémantique des arguments transmis au programme runmqakm a échoué.
202	L'objet identifié dans la commande n'est pas un objet reconnu.

Code d'erreur	Message d'erreur
203	L'action transmise n'est pas une action -keydb connue.
204	L'action transmise n'est pas une action -cert connue.
205	L'action transmise n'est pas une action -certreq connue.
206	Une balise est manquante pour la commande demandée.
207	La valeur transmise avec la balise -version n'est pas une valeur reconnue.
208	La valeur transmise avec la balise -size n'est pas une valeur reconnue.
209	La valeur transmise avec la balise -dn n'est pas au format correct.
210	La valeur transmise avec la balise -format n'est pas une valeur reconnue.
211	Une erreur s'est produite lors de l'ouverture du fichier.
212	PKCS12 n'est pas pris en charge à ce stade.
213	Le jeton cryptographique pour lequel vous tentez de modifier le mot de passe n'est pas protégé par mot de passe.
214	PKCS12 n'est pas pris en charge à ce stade.
215	Le mot de passe entré ne correspond pas à la force minimale requise.
216	Le mode FIPS n'est pas disponible.
217	Le nombre de jours que vous avez entré comme date d'expiration est hors de la plage autorisée.
218	La force du mot de passe n'a pas satisfait aux exigences minimales.
219	Aucun certificat par défaut n'a été trouvé dans la base de données de clés demandée.
220	Un statut de confiance non valide a été détecté.
221	Un algorithme de signature non pris en charge a été détecté. A ce stade, seuls MD5 et SHA1 sont pris en charge.
222	PCKS11 n'est pas pris en charge pour cette opération particulière.
223	L'action transmise n'est pas une action aléatoire connue.
224	Une longueur inférieure à zéro n'est pas admise.
225	Lorsque vous utilisez la balise -strong, la longueur minimale du mot de passe est de 14 caractères.

Code d'erreur	Message d'erreur
226	Lorsque vous utilisez la balise -strong, la longueur maximale du mot de passe est de 300 caractères.
227	L'algorithme MD5 n'est pas pris en charge en mode FIPS.
228	La balise de site n'est pas prise en charge pour la commande -cert -list. Cet attribut est ajouté à des fins de compatibilité avec les versions antérieures et d'amélioration future potentielle.
229	La valeur associée à la balise -ca n'est pas reconnue. La valeur doit être'true'ou'false'.
230	La valeur transmise avec la balise -type n'est pas valide.
231	La valeur transmise avec la balise -expire est inférieure à la plage autorisée.
232	L'algorithme de chiffrement utilisé ou demandé n'est pas pris en charge.
233	La cible existe déjà.

Protection des détails d'authentification de la base de données

Si vous utilisez l'authentification par nom d'utilisateur et mot de passe pour vous connecter au gestionnaire de base de données, vous pouvez les stocker dans le magasin de données d'identification MQ XA afin d'éviter de stocker le mot de passe en texte en clair dans le fichier `qm.ini` .

Mettez à jour XAOpenString pour le gestionnaire de ressources

Pour utiliser le magasin de données d'identification, vous devez modifier XAOpenString dans le fichier `qm.ini` . La chaîne est utilisée pour se connecter au gestionnaire de base de données. Vous spécifiez des zones remplaçables pour identifier où le nom d'utilisateur et le mot de passe sont remplacés dans la chaîne XAOpenString .

- La zone +USER+ est remplacée par la valeur de nom d'utilisateur stockée dans le magasin XACredentials.
- La zone +PASSWORD+ est remplacée par la valeur de mot de passe stockée dans le magasin XACredentials.

Les exemples suivants montrent comment modifier un XAOpenString afin d'utiliser le fichier de données d'identification pour se connecter à la base de données.

Connexion à une base de données Db2

```
XAResourceManager:
  Name=mydb2
  SwitchFile=db2swit
  XAOpenString=db=mydbname,uid=+USER+,pwd=+PASSWORD+,toc=t
  ThreadOfControl=THREAD
```

Connexion à une base de données Oracle

```
XAResourceManager:
  Name=myoracle
  SwitchFile=oraswit
  XAOpenString=Oracle_XA+Acc=P/+USER+/+PASSWORD++SesTm=35
```

```
            +LogDir=/tmp+threads=true
  ThreadOfControl=THREAD
```

Utilisation des données d'identification de la base de données dans le magasin de données d'identification XA MQ

Après avoir mis à jour le fichier `qm.ini` avec les chaînes de données d'identification remplaçables, vous devez ajouter le nom d'utilisateur et le mot de passe au magasin de données d'identification MQ à l'aide de la commande **setmqxacred** . Vous pouvez également utiliser **setmqxacred** pour modifier des données d'identification existantes, supprimer des données d'identification ou répertorier des données d'identification. Les exemples suivants présentent des cas d'utilisation typiques:

Ajout de données d'identification

La commande suivante sauvegarde de manière sécurisée le nom d'utilisateur et le mot de passe du gestionnaire de files d'attente `QM1` pour la ressource `mqdb2`.

```
setmqxacred -m QM1 -x mydb2 -u user1 -p Password2
```

Mise à jour des droits

Pour mettre à jour le nom d'utilisateur et le mot de passe utilisés pour la connexion à une base de données, émettez à nouveau la commande **setmqxacred** avec le nouveau nom d'utilisateur et le nouveau mot de passe:

```
setmqxacred -m QM1 -x mydb2 -u user3 -p Password4
```

Vous devez redémarrer le gestionnaire de files d'attente pour que les modifications soient prises en compte.

Suppression des données d'identification

La commande suivante supprime les données d'identification:

```
setmqxacred -m QM1 -x mydb2 -d
```

Liste des données d'identification

La commande suivante répertorie les données d'identification:

```
setmqxacred -m QM1 -l
```

Référence associée

setmqxacred

Sécurisation de Managed File Transfer

Juste après l'installation et si vous n'avez apporté aucune modification, Managed File Transfer présente un niveau de sécurité pouvant être adapté à des fins de test ou d'évaluation dans un environnement protégé. Toutefois, dans un environnement de production, vous devez envisager de contrôler de façon appropriée les utilisateurs pouvant démarrer des opérations de transfert de fichier et lire et écrire les fichiers transférés, et déterminer comment protéger l'intégrité des fichiers.

Tâches associées

Restriction des droits de groupe pour les ressources spécifiques à MFT

Droits de gestion pour les ressources spécifiques à MFT

«Utilisation de Advanced Message Security avec Managed File Transfer», à la page 623

Ce scénario explique comment configurer Advanced Message Security pour fournir la confidentialité des messages pour les données envoyées via un Managed File Transfer.

Référence associée

Droits d'accès de MFT aux systèmes de fichiers

Propriété commandPath MFT

Droits de publication du journal et des messages d'état des agents MFT

Authentification de connexion MFT et IBM MQ

L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

Méthodes de fourniture des détails des données d'identification

De nombreuses commandes Managed File Transfer prennent en charge les méthodes suivantes pour fournir les détails des données d'identification:

Détails fournis par les arguments de ligne de commande.
Les détails des données d'identification peuvent être spécifiés à l'aide des paramètres **-mquserid** et **-mqpassword** . Si le **-mqpassword** n'est pas fourni, l'utilisateur est invité à indiquer le mot de passe dans lequel l'entrée n'est pas affichée.

Détails fournis depuis un fichier de données d'identification : `MQMFTCredentials.xml`.
Les données d'identification détaillées peuvent être prédéfinies dans un fichier `MQMFTCredentials.xml` sous forme de texte en clair ou brouillé.

Pour plus d'informations sur la configuration d'un fichier MQMFTCredentials.xml sur IBM MQ for Multiplatforms , voir «Configuration de MQMFTCredentials.xml sur plusieurs plateformes», à la page 560.

Pour plus d'informations sur la configuration d'un fichier MQMFTCredentials.xml sur IBM MQ for z/OS , voir «Configuration de MQMFTCredentials.xml sur z/OS», à la page 561.

Priorité

L'ordre de priorité des méthodes pour déterminer les données d'identification détaillées est le suivant :

1. Argument de ligne de commande
2. Index `MQMFTCredentials.xml` par gestionnaire de files d'attente associé et utilisateur exécutant la commande
3. Index `MQMFTCredentials.xml` par gestionnaire de files d'attente associé
4. Mode de compatibilité amont par défaut dans lequel aucun détail de données d'identification n'est fourni pour permettre la compatibilité avec les versions précédentes de IBM MQou IBM WebSphere MQ

Remarques :

- Les commandes **`fteStartAgent`** et **`fteStartLogger`** ne prennent pas en charge l'argument de ligne de commande **-mquserid** ou **-mqpassword**, et les données d'identification détaillées ne peuvent être spécifiées qu'à l'aide du fichier `MQMFTCredentials.xml`.
- z/OS

  Sous z/OS, le mot de passe doit être en majuscules, même si le mot de passe de l'utilisateur est en minuscules. Par exemple, si le mot de passe de l'utilisateur est "motdepasse", il doit être entré sous la forme "MOTDEPASSE".

Référence associée

Quelle commande MFT se connecte à quel gestionnaire de files d'attente

Configuration de `MQMFTCredentials.xml` sur plusieurs plateformes

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert toutes les commandes MFT qui se connectent à un gestionnaire de files d'attente pour fournir les données d'identification par ID utilisateur et mot de passe. De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT .

Pourquoi et quand exécuter cette tâche

Les éléments du fichier `MQMFTCredentials.xml` doivent être conformes au schéma `MQMFTCredentials.xsd` . Pour plus d'informations sur le format de `MQMFTCredentials.xml`, voir Format de fichier des données d'identification MFT.

Vous trouverez un exemple de fichier de données d'identification dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials` .

Vous pouvez disposer d'un fichier de données d'identification MFT pour le gestionnaire de files d'attente de coordination, d'un fichier pour le gestionnaire de files d'attente de commandes, d'un fichier pour chaque agent et d'un fichier pour chaque consignateur. Vous pouvez également disposer d'un fichier qui est utilisé par tous les éléments de votre topologie.

L'emplacement par défaut du fichier de données d'identification MFT est le suivant:

UNIX and Linux

$HOME

Windows

%USERPROFILE% ou %HOMEDRIVE%%HOMEPATH%

Si le fichier de données d'identification est stocké à un autre emplacement, vous pouvez utiliser les propriétés suivantes pour indiquer où les commandes doivent le rechercher:

Tableau 91. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour diverses commandes.

Type de commande	Fichier de propriétés	Nom de la propriété
Commande qui se connecte au gestionnaire de file d'attente de coordination	coordination.properties	coordinationQMgrAuthenticationCredentialsFile
Commande qui se connecte au gestionnaire de files d'attente de commandes	connection.properties	connectionQMgrAuthenticationCredentialsFile
Commande qui se connecte à un processus d'agent	agent.properties	agentQMgrAuthenticationCredentialsFile
Commande qui se connecte à un processus de consignateur	logger.properties	loggerQMgrAuthenticationCredentialsFile

Tableau 92. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour les agents et les processus de consignateur.		
Type de commande	**Fichier de propriétés**	**Nom de la propriété**
Agents MFT	agent.properties	agentQMgrAuthenticationCredentialsFile
MFT loggers	logger.properties	loggerQMgrAuthenticationCredentialsFile

Pour plus de détails sur les commandes et les processus qui se connectent à quel gestionnaire de files d'attente, voir Quelles commandes et quels processus MFT se connectent à quel gestionnaire de files d'attente.

Etant donné que le fichier de données d'identification contient des informations d'ID utilisateur et de mot de passe, il requiert des droits spéciaux pour empêcher tout accès non autorisé à ce fichier:

Linux UNIX **UNIX and Linux**

```
chown <agent owner userid>
chmod 600
```

Windows **Windows**

Vérifiez que l'héritage n'est pas activé, puis supprimez tous les ID utilisateur à l'exception de ceux qui exécutent l'agent ou le consignateur qui utiliseront le fichier de données d'identification.

Les données d'identification utilisées pour la connexion à un gestionnaire de files d'attente de coordination MFT , dans le plug-in IBM MQ Explorer Managed File Transfer , dépendent du type de configuration:

Global (configuration sur disque local)

Une configuration globale utilise le fichier de données d'identification spécifié dans les propriétés de coordination et de commande.

Local (défini dans IBM MQ Explorer):

Une configuration locale utilise les propriétés des détails de connexion du gestionnaire de files d'attente associé dans IBM MQ Explorer.

Tâches associées

«Activation de l'authentification de connexion pour MFT», à la page 563

L'authentification de connexion du plug-in IBM MQ Explorer MFT se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes et l'authentification de connexion pour un agent Managed File Transfer se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes peuvent être exécutées en mode compatibilité ou en mode d'authentification MQCSP.

Référence associée

Format du fichier de données d'identification MFT

`fteObfuscate`: chiffrement des données sensibles

z/OS Configuration de `MQMFTCredentials.xml` sur z/OS

Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert tous les agents MFT , ainsi que les commandes qui se connectent à un gestionnaire de files d'attente, pour fournir les données d'identification par ID utilisateur et mot de passe.

De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données.

Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT . Notez que les fichiers de données d'identification sont facultatifs, mais il est plus facile de définir le ou les fichiers dont vous avez besoin avant de personnaliser l'environnement.

En outre, si vous disposez de fichiers de données d'identification, vous recevez moins de messages d'avertissement. Les messages d'avertissement vous informent que MFT considère que la sécurité du gestionnaire de files d'attente est désactivée et que, par conséquent, vous ne fournissez pas de détails d'authentification.

Vous trouverez un exemple de fichier de données d'identification dans le répertoire `MQ_INSTALLATION_PATH/mqft/samples/credentials`.

Voici un exemple de fichier `MQMFTCredentials.xml`:

```
<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MFTCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MFTCredentials.xsd">
  <tns:qmgr name="MQPH" user="ADMIN" mqUserId="JOHNDOEH" mqPassword="cXXXX" />
  <tns:qmgr name="MQPI" user="ADMIN" mqUserId="JOHNDOEI" mqPassword="yXXXX" />
  <tns:qmgr name="MQPH" mqUserId="NONEH" mqPassword="yXXXX" />
  <tns:qmgr name="MQPI" mqUserId="NONEI" mqPassword="yXXXX" />
</tns:mqmftCredentials>
```

Lorsqu'un travail avec l'ID utilisateur ADMIN doit se connecter au gestionnaire de files d'attente MQPH, il transmet l'ID utilisateur *JOHNDOEH* et utilise le mot de passe *cXXXX*.

Si le travail est exécuté par un autre ID utilisateur et qu'il connecte MQPH, il transmet l'ID utilisateur *NONEH* et le mot de passe *yXXXX*.

L'emplacement par défaut du fichier `MQMFTCredentials.xml` est le répertoire de base de l'utilisateur sur z/OS UNIX System Services (USS). Il est également possible de stocker le fichier dans un autre emplacement sur USS ou dans un membre d'un fichier partitionné.

Si le fichier de données d'identification est stocké à un autre emplacement, vous pouvez utiliser les propriétés suivantes pour indiquer où les commandes doivent le rechercher:

Tableau 93. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour diverses commandes.

Type de commande	Fichier de propriétés	Nom de la propriété
Commande qui se connecte au gestionnaire de file d'attente de coordination	coordination.properties	coordinationQMgrAuthenticationCredentialsFile
Commande qui se connecte au gestionnaire de files d'attente de commandes	connection.properties	connectionQMgrAuthenticationCredentialsFile
Commande qui se connecte à un processus d'agent	agent.properties	agentQMgrAuthenticationCredentialsFile
Commande qui se connecte à un processus de consignateur	logger.properties	loggerQMgrAuthenticationCredentialsFile

Tableau 94. : Propriétés qui définissent l'emplacement du fichier MQMFTCredentials.xml pour les agents et les processus de consignateur.

Type de commande	Fichier de propriétés	Nom de la propriété
Agents MFT	agent.properties	agentQMgrAuthenticationCredentialsFile
MFT loggers	logger.properties	loggerQMgrAuthenticationCredentialsFile

Pour plus de détails sur les commandes et les processus qui se connectent à quel gestionnaire de files d'attente, voir Quelles commandes et quels processus MFT se connectent à quel gestionnaire de files d'attente.

Pour créer le fichier de données d'identification dans un fichier partitionné, procédez comme suit:

- Créez un ensemble de données partitionnées étendu avec le format VB et la longueur d'enregistrement logique (Lrecl) 200.
- Créez un membre dans le fichier, notez le fichier et le membre et ajoutez le code suivant au membre:

```
<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MQMFTCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MQMFTCredentials.xsd">
  <!—-credentials information goes here-->
</tns:mqmftCredentials>
```

Vous pouvez protéger le fichier de données d'identification à l'aide d'un produit de sécurité, par exemple RACF, mais les ID utilisateur exécutant les commandes Managed File Transfer et administrant les processus d'agent et de consignateur ont besoin d'un accès en lecture à ce fichier.

Vous pouvez masquer les informations de ce fichier à l'aide du JCL du membre BFGCROBS. Cette opération utilise le fichier et chiffre l'ID utilisateur et le mot de passe IBM MQ . Par exemple, le membre BFGCROBS prend la ligne

```
<tns:qmgr name="MQPI" user="JOHNDOE2" mqUserId="JOHNDOE1" mqPassword="yXXXX" />
```

et crée

```
<tns:qmgr mqPasswordCipher="e977c61e9b9c363c" mqUserIdCipher="c394c5887867157c"
name="MQPI" user="JOHNDOE2"/>
```

Si vous souhaitez conserver le mappage de l'ID utilisateur à l'ID utilisateur IBM MQ , vous pouvez ajouter des commentaires au fichier. Par exemple

```
<!--      name="MQPI" user="ADMIN"    mqUserId="JOHNDOE1 -->
```

Ces commentaires ne sont pas modifiés par le processus d'obscurcissement.

Notez que le contenu est masqué et qu'il n'est pas fortement chiffré. Vous devez limiter les ID utilisateur qui ont accès au fichier.

Tâches associées

«Configuration de MQMFTCredentials.xml sur plusieurs plateformes», à la page 560
Si Managed File Transfer (MFT) est configuré avec la sécurité activée, l'authentification de connexion requiert toutes les commandes MFT qui se connectent à un gestionnaire de files d'attente pour fournir les données d'identification par ID utilisateur et mot de passe. De même, les consignateurs MFT peuvent être tenus de spécifier un ID utilisateur et un mot de passe lors de la connexion à une base de données. Ces données d'identification peuvent être stockées dans le fichier de données d'identification MFT .

Activation de l'authentification de connexion pour MFT

L'authentification de connexion du plug-in IBM MQ Explorer MFT se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes et l'authentification de connexion pour un agent Managed File Transfer se connectant à un gestionnaire de files d'attente de coordination ou à un gestionnaire de files d'attente de commandes peuvent être exécutées en mode compatibilité ou en mode d'authentification MQCSP.

Pourquoi et quand exécuter cette tâche

Avant IBM MQ 9.1.1, le mode de compatibilité est le paramètre par défaut pour l'authentification de connexion. Toutefois, vous pouvez désactiver le mode de compatibilité par défaut et activer le mode d'authentification MQCSP.

V 9.1.1 Depuis IBM MQ 9.1.1, le mode d'authentification MQCSP est le mode par défaut.

Pour l'authentification de connexion pour le plug-in IBM MQ Explorer Managed File Transfer ou pour les agents Managed File Transfer qui se connectent à un gestionnaire de files d'attente à l'aide du transport CLIENT, les mots de passe de plus de 12 caractères sont uniquement pris en charge pour le mode d'authentification MQCSP. Si vous indiquez un mot de passe de plus de 12 caractères lors de l'autorisation en mode compatibilité, une erreur se produit et l'agent ne s'authentifie pas auprès du gestionnaire de files d'attente. Voir le message BFGAG0187E dans Messages de diagnostic: BFGAG0001 - BFGAG9999.

Procédure

- Pour sélectionner le mode d'authentification de connexion pour un gestionnaire de files d'attente de coordination ou un gestionnaire de files d'attente de commandes dans IBM MQ Explorer, procédez comme suit:
  a) Sélectionnez le gestionnaire de files d'attente auquel vous souhaitez vous connecter.
  b) Cliquez avec le bouton droit de la souris et sélectionnez **Détails de connexion-> Propriétés** dans le menu contextuel.
  c) Cliquez sur l'onglet **ID utilisateur**.
  d) Vérifiez que la case correspondant au mode d'authentification de connexion que vous souhaitez utiliser est cochée:
     - V 9.1.0 Depuis IBM MQ 9.1.0, par défaut, la case **User identification compatibility mode** est désélectionnée. Cela signifie que si la case **Activer l'identification de l'utilisateur** est cochée, IBM MQ Explorer utilisera l'authentification MQCSP lors de la connexion au gestionnaire de files d'attente. Si IBM MQ Explorer doit se connecter au gestionnaire de files d'attente à l'aide du mode de compatibilité au lieu de l'authentification MQCSP, vérifiez que les cases à cocher **Activer l'identification de l'utilisateur** et **Mode de compatibilité d'identification de l'utilisateur** sont sélectionnées.
     - Avant IBM MQ 9.1.0, par défaut, la case **User identification compatibility mode** est cochée. Cela signifie que si la case **Activer l'identification de l'utilisateur** est cochée, IBM MQ Explorer utilisera le mode compatibilité lors de la connexion au gestionnaire de files d'attente. Si IBM MQ Explorer doit se connecter au gestionnaire de files d'attente à l'aide de l'authentification MQCSP, vérifiez que la case **Activer l'identification de l'utilisateur** est cochée et que la case **Mode de compatibilité d'identification de l'utilisateur** est décochée.
- Pour activer ou désactiver le mode d'authentification MQCSP pour un agent Managed File Transfer à l'aide du fichier `MQMFTCredentials.xml` , ajoutez le paramètre **useMQCSPAuthentication** au fichier `MQMFTCredentials.xml` pour l'utilisateur approprié.

  Le paramètre **useMQCSPAuthentication** a les valeurs suivantes:

  true
  Le mode d'authentification MQCSP permet d'authentifier l'utilisateur auprès du gestionnaire de files d'attente.

  V 9.1.1 Depuis IBM MQ 9.1.1, `true` est la valeur par défaut. Si le paramètre **useMQCSPAuthentication** n'est pas spécifié, il est défini par défaut sur `true` et le mode d'authentification MQCSP est utilisé pour authentifier l'utilisateur avec le gestionnaire de files d'attente.

  false
  Le mode compatibilité est utilisé pour authentifier l'utilisateur auprès du gestionnaire de files d'attente.

  Avant IBM MQ 9.1.1, si le paramètre **useMQCSPAuthentication** n'est pas spécifié, il est défini par défaut sur `false` et le mode de compatibilité est utilisé pour authentifier l'utilisateur avec le gestionnaire de files d'attente.

L'exemple suivant montre comment définir le paramètre **useMQCSPAuthentication** dans le fichier `MQMFTCredentials.xml` :

```
<tns:qmgr name="CoordQueueMgr" user="ernest" mqUserId="ernest"
    mqPassword="AveryL0ngPassw0rd2135" useMQCSPAuthentication="true"/>
```

Concepts associés

«Protection par mot de passe MQCSP», à la page 30
Depuis IBM MQ 8.0, vous pouvez envoyer des mots de passe inclus dans la structure MQCSP qui sont protégés, à l'aide de la fonctionnalité IBM MQ , ou chiffrés, à l'aide du chiffrement TLS.

Référence associée

«Authentification de connexion MFT et IBM MQ», à la page 559
L'authentification de connexion permet à un gestionnaire de files d'attente d'être configuré pour authentifier les applications à l'aide d'un ID utilisateur et d'un mot de passe fournis. Si la sécurité est activée pour le gestionnaire de files d'attente associé et que les données d'identification (ID utilisateur et mot de passe) sont requises, la fonction d'authentification de connexion doit être activée pour que la connexion à un gestionnaire de files d'attente puisse être établie. L'authentification de connexion peut être exécutée en mode compatibilité ou en mode d'authentification MQCSP.

Format du fichier de données d'identification MFT

MFT bacs à sable

Vous pouvez restreindre la zone du système de fichiers à laquelle l'agent peut accéder dans le cadre d'un transfert. La zone à laquelle l'agent est limité est appelée le bac à sable. Vous pouvez appliquer des restrictions à l'agent ou à l'utilisateur qui demande un transfert.

Les bacs à sable ne sont pas pris en charge lorsque l'agent est un agent de pont de protocole ou un agent de pont Connect:Direct . Vous ne pouvez pas utiliser le bac à sable d'agent pour les agents qui doivent effectuer un transfert vers ou depuis des files d'attente IBM MQ .

Référence associée

«Utilisation des bacs à sable d'agent MFT», à la page 565
Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

«Utilisation des bacs à sable utilisateur MFT», à la page 567
Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

Utilisation des bacs à sable d'agent MFT

Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

Vous ne pouvez pas utiliser le bac à sable d'agent pour les agents qui sont transférés vers ou depuis des files d'attente IBM MQ . La restriction de l'accès aux files d'attente IBM MQ avec bac à sable peut être implémentée à la place en utilisant le bac à sable utilisateur, qui est la solution recommandée pour toutes les exigences en matière de bac à sable. Pour plus d'informations sur le bac à sable utilisateur, voir «Utilisation des bacs à sable utilisateur MFT», à la page 567

Pour activer le bac à sable de l'agent, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez restreindre:

```
sandboxRoot=[!]restricted_directory_nameseparator...separator[!]restricted_directory_name
```

où :

- *restricted_directory_name* est un chemin de répertoire à autoriser ou à refuser.

- ! est facultatif et indique que la valeur suivante pour *restricted_directory_name* est refusée (exclue). Si ! n'est pas spécifié, *restricted_directory_name* est un chemin d'accès autorisé (inclus).
- *separator* est le séparateur spécifique à la plateforme.

Par exemple, si vous souhaitez restreindre l'accès de AGENT1 au répertoire `/tmp` uniquement, mais ne pas autoriser l'accès au sous-répertoire `private` , définissez la propriété comme suit dans le fichier `agent.properties` appartenant à AGENT1: `sandboxRoot=/tmp:!/tmp/private`.

La propriété sandboxRoot est décrite dans Propriétés avancées de l'agent.

Les bacs à sable d'agent et d'utilisateur ne sont pas pris en charge sur les agents de pont de protocole ou sur les agents de pont Connect:Direct .

Utilisation d'un bac à sable sur les plateformes UNIX, Linuxet Windows

ULW Sur les plateformes UNIX, Linuxet Windows , le bac à sable restreint les répertoires dans lesquels un Managed File Transfer Agent peut effectuer des opérations de lecture et d'écriture. Lorsque le bac à sable est activé, le Managed File Transfer Agent peut lire et écrire dans les répertoires spécifiés comme étant autorisés, ainsi que dans les sous-répertoires que les répertoires spécifiés contiennent, sauf si les sous-répertoires sont spécifiés comme étant refusés dans sandboxRoot. Le bac à sable Managed File Transfer n'est pas prioritaire sur la sécurité du système d'exploitation. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer de l'accès de niveau système d'exploitation approprié à n'importe quel répertoire pour pouvoir lire ou écrire dans le répertoire. Un lien symbolique vers un répertoire n'est pas suivi si le répertoire auquel il est lié se trouve en dehors des répertoires sandboxRoot spécifiés (et des sous-répertoires).

Utilisation d'un bac à sable sous z/OS

z/OS Sous z/OS, le bac à sable restreint les qualificateurs de nom de fichier que le Managed File Transfer Agent peut lire et dans lesquels il peut écrire. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer des droits d'accès appropriés au système d'exploitation pour tous les fichiers concernés. Si vous placez une valeur de qualificateur de nom de fichier sandboxRoot entre guillemets, la valeur suit la convention z/OS normale et est traitée comme qualifiée complète. Si vous omettez les guillemets, sandboxRoot est préfixé avec l'ID utilisateur en cours. Par exemple, si vous définissez la propriété sandboxRoot sur la valeur suivante: `sandboxRoot=//test`, l'agent peut accéder aux ensembles de données suivants (en notation z/OS standard) `//`*`username`*`.test.**` Au moment de l'exécution, si les niveaux initiaux du nom de fichier entièrement résolu ne correspondent pas à sandboxRoot, la demande de transfert est rejetée.

Utilisation d'un bac à sable sur des systèmes IBM i

IBM i Pour les fichiers du système de fichiers intégré sur les systèmes IBM i , le bac à sable restreint les répertoires dans lesquels un Managed File Transfer Agent peut effectuer des opérations de lecture et d'écriture. Lorsque le bac à sable est activé, le Managed File Transfer Agent peut lire et écrire dans les répertoires spécifiés comme étant autorisés, ainsi que dans les sous-répertoires que les répertoires spécifiés contiennent, sauf si les sous-répertoires sont spécifiés comme étant refusés dans sandboxRoot. Le bac à sable Managed File Transfer n'est pas prioritaire sur la sécurité du système d'exploitation. L'utilisateur qui a démarré Managed File Transfer Agent doit disposer de l'accès de niveau système d'exploitation approprié à n'importe quel répertoire pour pouvoir lire ou écrire dans le répertoire. Un lien symbolique vers un répertoire n'est pas suivi si le répertoire auquel il est lié se trouve en dehors des répertoires sandboxRoot spécifiés (et des sous-répertoires).

Référence associée

«Vérifications supplémentaires pour les transferts de caractères génériques», à la page 570
Si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez indiquer que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

«Utilisation des bacs à sable d'agent MFT», à la page 565
Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

Le fichier MFT agent.properties

Utilisation des bacs à sable utilisateur MFT

Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

Les bacs à sable utilisateur ne sont pas pris en charge lorsque l'agent est un agent de pont de protocole ou un agent de pont Connect:Direct .

Pour activer le bac à sable utilisateur, ajoutez la propriété suivante au fichier agent.properties de l'agent que vous souhaitez restreindre:

```
userSandboxes=true
```

Lorsque cette propriété est présente et définie sur true, l'agent utilise les informations du fichier *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name*/agents/*agent_name*/UserSandboxes.xml pour déterminer les parties du système de fichiers auxquelles l'utilisateur qui demande le transfert peut accéder.

Le XML UserSandboxes.xml est composé d'un élément <agent> qui contient zéro ou plusieurs éléments <sandbox> . Ces éléments décrivent quelles règles sont appliquées à quels utilisateurs. L'attribut user de l'élément <sandbox> est un modèle utilisé pour établir une correspondance avec l'utilisateur MQMD de la demande.

Le fichier UserSandboxes.xml est rechargé périodiquement par l'agent et toute modification valide apportée au fichier aura une incidence sur le comportement de l'agent. L'intervalle de rechargement par défaut est de 30 secondes. Vous pouvez modifier cet intervalle en spécifiant la propriété d'agent xmlConfigReloadInterval dans le fichier agent.properties .

Si vous spécifiez l'attribut ou la valeur userPattern="regex" , l'attribut user est interprété comme une expression régulière Java . Pour plus d'informations, voir Expressions régulières utilisées par MFT.

Si vous ne spécifiez pas l'attribut ou la valeur userPattern="regex" , l'attribut user est interprété comme un modèle avec les caractères génériques suivants:

- astérisque (*), qui représente zéro ou plusieurs caractères
- point d'interrogation (?), qui représente exactement un caractère

Les correspondances sont effectuées dans l'ordre dans lequel les éléments <sandbox> sont répertoriés dans le fichier. Seule la première correspondance est utilisée, toutes les correspondances potentielles suivantes dans le fichier sont ignorées. Si aucun des éléments <sandbox> spécifiés dans le fichier ne correspond à l'utilisateur MQMD associé au message de demande de transfert, le transfert ne peut pas accéder au système de fichiers. Lorsqu'une correspondance a été trouvée entre le nom d'utilisateur MQMD et un attribut user , la correspondance identifie un ensemble de règles dans un élément <sandbox> qui sont appliquées au transfert. Cet ensemble de règles est utilisé pour déterminer quels fichiers, ou ensembles de données, peuvent être lus ou écrits dans le cadre du transfert.

Chaque ensemble de règles peut spécifier un élément <read> , qui identifie les fichiers qui peuvent être lus, et un élément <write> , qui identifie les fichiers qui peuvent être écrits. Si vous omettez les éléments <read> ou <write> d'un ensemble de règles, il est supposé que l'utilisateur associé à cet ensemble de règles n'est pas autorisé à effectuer des lectures ou des écritures, selon le cas.

Remarque : L'élément <read> doit être antérieur à l'élément <write> et l'élément <include> doit être antérieur à l'élément <exclude> , dans le fichier UserSandboxes.xml .

Chaque élément <read> ou <write> contient un ou plusieurs canevas utilisés pour déterminer si un fichier se trouve dans le bac à sable et peut être transféré. Spécifiez ces modèles à l'aide des éléments <include> et <exclude> . L'attribut name de l'élément <include> ou <exclude> spécifie le modèle

à mettre en correspondance. Un attribut `type` facultatif indique si la valeur de nom est un fichier ou un modèle de file d'attente. Si l'attribut `type` n'est pas spécifié, l'agent traite le modèle comme un modèle de chemin de fichier ou de répertoire. Exemple :

```
<tns:read>
    <tns:include name="/home/user/**"/>
    <tns:include name="USER.**" type="queue"/>
    <tns:exclude name="/home/user/private/**"/>
</tns:read>
```

Les modèles `<include>` et `<exclude>` `name` sont utilisés par l'agent pour déterminer si les fichiers, les ensembles de données, ou les files d'attente peuvent être lus ou écrits. Une opération est autorisée si le chemin de fichier canonique, l'ensemble de données ou le nom de file d'attente correspond à au moins un des modèles inclus et à exactement zéro des modèles exclus. Les modèles spécifiés à l'aide de l'attribut `name` des éléments `<include>` et `<exclude>` utilisent les séparateurs de chemin et les conventions appropriés à la plateforme sur laquelle l'agent s'exécute. Si vous spécifiez des chemins de fichier relatifs, les chemins sont résolus par rapport à la propriété `transferRoot` de l'agent.

Lors de la spécification d'une restriction de file d'attente, la syntaxe QUEUE@QUEUEMANAGER est prise en charge, avec les règles suivantes:

- Si le caractère at (@) est manquant dans l'entrée, le modèle est traité comme un nom de file d'attente accessible sur n'importe quel gestionnaire de files d'attente. Par exemple, si le modèle est `name` , il est traité de la même manière que `name@**`.
- Si le caractère arobase (@) est le premier caractère de l'entrée, le modèle est traité comme un nom de gestionnaire de files d'attente et toutes les files d'attente du gestionnaire de files d'attente sont accessibles. Par exemple, si le modèle est `@name` , il est traité de la même manière que `**@name`.

Les caractères génériques suivants ont une signification spéciale lorsque vous les spécifiez dans le cadre de l'attribut `name` des éléments `<include>` et `<exclude>` :

*****
: Un astérisque unique correspond à zéro ou plusieurs caractères dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

?
: Un point d'interrogation correspond exactement à un caractère dans un nom de répertoire ou dans un qualificateur d'un nom de fichier ou d'un nom de file d'attente .

******
: Deux astérisques correspondent à zéro ou plusieurs noms de répertoire, ou à zéro ou plusieurs qualificatifs dans un nom de fichier ou un nom de file d'attente . En outre, les chemins qui se terminent par un séparateur de chemin ont un "**" implicite ajouté à la fin du chemin. Ainsi, `/home/user/` est identique à `/home/user/**`.

Exemple :

- `/**/test/**` correspond à tout fichier dont le chemin contient un répertoire `test`
- `/test/file?` correspond à tout fichier du répertoire `/test` qui commence par la chaîne `file` suivie d'un caractère unique
- `c:\test\*.txt` correspond à tout fichier du répertoire `c:\test` avec une extension `.txt`
- `c:\test\**\*.txt` correspond à n'importe quel fichier du répertoire `'c:\test` ou à l'un de ses sous-répertoires dont l'extension est `.txt`
- z/OS `//'TEST.*.DATA'` correspond à tout fichier dont le premier qualificateur est TEST, dont le second est un qualificateur et dont le troisième est DATA.
- *@QM1 correspond à toute file d'attente du gestionnaire de files d'attente QM1 comportant un qualificateur unique.

- TEST.*.QUEUE@QM1 correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1 qui possède le premier qualificateur TEST, un deuxième qualificateur et un troisième qualificateur QUEUE.
- **@QM1 correspond à n'importe quelle file d'attente du gestionnaire de files d'attente QM1.

Liens symboliques

Vous devez résoudre complètement tous les liens symboliques que vous utilisez dans les chemins de fichier du fichier UserSandboxes.xml en spécifiant des liens fixes dans les éléments <include> et <exclude> . Par exemple, si vous disposez d'un lien symbolique dans lequel /var est mappé à /SYSTEM/var, vous devez spécifier ce chemin en tant que <tns:include name="/SYSTEM/var"/>, sinon le transfert prévu échoue avec une erreur de sécurité du bac à sable de l'utilisateur.

Exemple

Cet exemple montre comment autoriser l'utilisateur avec le nom d'utilisateur MQMD guest à transférer un fichier depuis le répertoire /home/user/public ou l'un de ses sous-répertoires sur le système où l'agent AGENT_JUPITER est en cours d'exécution, en ajoutant l'élément <sandbox> suivant au fichier UserSandboxes.xml dans le répertoire de configuration d'AGENT_JUPITER:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
        xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
    <tns:agent>
        <tns:sandbox user="guest">
            <tns:read>
                <tns:include name="/home/user/public/**"/>
             </tns:read>
        </tns:sandbox>
    </tns:agent>
</tns:userSandboxes>
```

Exemple

Cet exemple montre comment autoriser tout utilisateur avec le nom d'utilisateur MQMD account suivi d'un chiffre unique, par exemple account4, à effectuer les actions suivantes:

- Transférez tout fichier à partir du répertoire /home/account ou de l'un de ses sous-répertoires, à l'exception du répertoire /home/account/private sur le système où l'agent AGENT_SATURN est en cours d'exécution
- Transférez tout fichier dans le répertoire /home/account/output ou dans l'un de ses sous-répertoires sur le système où l'agent AGENT_SATURN est en cours d'exécution.
- Lire les messages des files d'attente du gestionnaire de files d'attente local en commençant par le préfixe ACCOUNT. , sauf s'il commence par ACCOUNT.PRIVATE. (c'est-à-dire avec PRIVATE au deuxième niveau).
- Transférez les données dans les files d'attente en commençant par le préfixe ACCOUNT.OUTPUT. sur n'importe quel gestionnaire de files d'attente.

Pour permettre à un utilisateur doté du nom d'utilisateur MQMD account d'effectuer ces actions, ajoutez l'élément <sandbox> suivant au fichier UserSandboxes.xml, dans le répertoire de configuration d'AGENT_SATURN:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
            xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
            xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
    <tns:agent>
        <tns:sandbox user="account[0-9]" userPattern="regex">
            <tns:read>
                <tns:include name="/home/account/**"/>
```

```
                <tns:include name="ACCOUNT.**" type="queue"/>
                <tns:exclude name="ACCOUNT.PRIVATE.**" type="queue"/>
                <tns:exclude name="/home/account/private/**"/>
                                                        </tns:read>
            <tns:write>
                <tns:include name="/home/account/output/**"/>
                <tns:include name="ACCOUNT.OUTPUT.**" type="queue"/>
            </tns:write>
        </tns:sandbox>
    </tns:agent>
</tns:userSandboxes>
```

Référence associée

«Vérifications supplémentaires pour les transferts de caractères génériques», à la page 570
Si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez indiquer que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

Le fichier MFT `agent.properties`

Vérifications supplémentaires pour les transferts de caractères génériques

Si un agent a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers, vous pouvez indiquer que des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour cet agent.

Propriété additionalWildcardSandboxChecking

Pour activer une vérification supplémentaire des transferts de caractères génériques, ajoutez la propriété suivante au fichier `agent.properties` de l'agent que vous souhaitez vérifier.

```
additionalWildcardSandboxChecking=true
```

Lorsque cette propriété est définie sur true et que l'agent effectue une demande de transfert qui tente de lire un emplacement qui se trouve en dehors du bac à sable défini pour la correspondance de fichier du caractère générique, le transfert échoue. S'il existe plusieurs transferts dans une même demande de transfert et que l'une de ces demandes échoue en raison d'une tentative de lecture d'un emplacement en dehors du bac à sable, le transfert complet échoue. Si la vérification échoue, la cause de l'échec est indiquée dans un message d'erreur.

Si la propriété additionalWildcardSandboxChecking est omise du fichier `agent.properties` d'un agent ou est définie sur false, aucune vérification supplémentaire n'est effectuée sur les transferts de caractères génériques pour cet agent.

Messages d'erreur pour la vérification des caractères génériques

Les messages qui sont signalés lorsqu'une demande de transfert générique est effectuée vers un emplacement en dehors d'un emplacement de bac à sable configuré sont les suivants.

Le message suivant apparaît lorsqu'un chemin de fichier générique dans une demande de transfert se trouve en dehors du bac à sable restreint:

```
BFGSS0077E: La tentative de lecture du chemin d'accès au fichier chemin a été refusée.
Le chemin d'accès au fichier se trouve hors du bac à sable de transfert restreint.
```

Le message suivant se produit lorsqu'un transfert au sein d'une demande de transfert multiple contient une demande de transfert générique dans laquelle le chemin se trouve en dehors du bac à sable restreint:

```
BFGSS0078E: La tentative de lecture du chemin d'accès au fichier: chemin a été ignorée car un
autre transfert
dans le transfert géré, tentative de lecture en dehors du bac à sable de transfert restreint.
```

Le message suivant s'affiche lorsqu'un fichier se trouve en dehors du bac à sable restreint:

```
BFGSS0079E: La tentative de lecture du fichier chemin d'accès au fichier a été refusée.
Le fichier se trouve en dehors du bac à sable de transfert restreint.
```

Le message suivant se produit dans une demande de transfert multiple où une autre demande de transfert générique a entraîné la non-prise en compte de cette demande:

```
BFGSS0080E: La tentative de lecture du fichier: chemin d'accès au fichier a été ignorée car un
autre transfert
dans le transfert géré, tentative de lecture en dehors du bac à sable de transfert restreint.
```

Dans le cas de transferts de fichiers uniques qui n'incluent pas de caractères génériques, le message qui est signalé lorsque le transfert implique un fichier situé en dehors du bac à sable est inchangé par rapport aux éditions précédentes:

```
Echec avec BFGIO0056E: La tentative de lecture du fichier "FILE" a été refusée.
Le fichier se trouve en dehors du bac à sable de transfert restreint.
```

Référence associée

«Utilisation des bacs à sable utilisateur MFT», à la page 567
Vous pouvez restreindre la zone du système de fichiers dans laquelle les fichiers peuvent être transférés en fonction du nom d'utilisateur MQMD qui demande le transfert.

«Utilisation des bacs à sable d'agent MFT», à la page 565
Pour ajouter un niveau de sécurité supplémentaire à Managed File Transfer, vous pouvez restreindre la zone d'un système de fichiers à laquelle un agent peut accéder.

Le fichier MFT `agent.properties`

Configuration du chiffrement SSL ou TLS pour MFT

Vous pouvez utiliser SSL ou TLS avec IBM MQ Managed File Transfer pour sécuriser les communications entre les agents et leurs gestionnaires de files d'attente d'agent, les commandes et les gestionnaires de files d'attente auxquels ils se connectent, ainsi que les différentes connexions entre les gestionnaires de files d'attente et les gestionnaires de files d'attente dans votre topologie.

Avant de commencer

Vous pouvez utiliser le chiffrement SSL ou TLS pour chiffrer les messages qui transitent par une topologie IBM MQ Managed File Transfer . Ces gestionnaires sont les suivants :

- Messages transmis entre un agent et son gestionnaire de files d'attente d'agent.
- Messages des commandes et des gestionnaires de files d'attente auxquels elles se connectent.
- Messages internes qui circulent entre les gestionnaires de files d'attente d'agent, les gestionnaires de files d'attente de commandes et le gestionnaire de files d'attente de coordination dans la topologie.

Pourquoi et quand exécuter cette tâche

Pour des informations générales sur l'utilisation de SSL avec IBM MQ, voir «Utilisation de SSL/TLS», à la page 282. En termes IBM MQ , Managed File Transfer est une application client Java standard.

Pour utiliser SSL avec Managed File Transfer, procédez comme suit:

Procédure

1. Créez un fichier de clés certifiées et éventuellement un fichier de clés (ces fichiers peuvent être identiques). Si vous n'avez pas besoin de l'authentification client (c'est-à-dire, SSLCAUTH=OPTIONAL sur les canaux), vous n'avez pas besoin de fournir un magasin de clés. Vous avez besoin d'un magasin de clés de confiance uniquement pour authentifier le certificat du gestionnaire de files d'attente.

   L'algorithme de clé utilisé pour créer des certificats pour le magasin de clés de confiance et les magasins de clés doit être RSA pour fonctionner avec IBM MQ.
2. Configurez votre gestionnaire de files d'attente IBM MQ pour utiliser SSL.

   Pour plus d'informations sur la configuration d'un gestionnaire de files d'attente pour utiliser SSL avec IBM MQ Explorer , par exemple, voir Configuration de SSL sur les gestionnaires de files d'attente.

3. Sauvegardez le fichier de clés certifiées et le fichier de clés (si vous en avez un) dans un emplacement approprié. Un emplacement suggéré est le répertoire *config_directory*/*coordination_qmgr*/agents/*agent_name* .
4. Définissez les propriétés SSL requises pour chaque gestionnaire de files d'attente SSL dans le fichier de propriétés Managed File Transfer approprié. Chaque ensemble de propriétés fait référence à un gestionnaire de files d'attente distinct (agent, coordination et commande), bien qu'un gestionnaire de files d'attente puisse exécuter deux ou plusieurs de ces rôles.

   L'une des propriétés **CipherSpec** ou **CipherSuite** est requise, sinon le client tente de se connecter sans SSL. Les propriétés **CipherSpec** ou **CipherSuite** sont fournies en raison des différences de terminologie entre IBM MQ et Java. Managed File Transfer accepte l'une ou l'autre propriété et effectue la conversion nécessaire. Vous n'avez donc pas besoin de définir les deux propriétés. Si vous spécifiez à la fois les propriétés **CipherSpec** ou **CipherSuite** , **CipherSpec** est prioritaire.

   La propriété **PeerName** est facultative. Vous pouvez définir la propriété sur le nom distinctif du gestionnaire de files d'attente auquel vous souhaitez vous connecter. Managed File Transfer rejette les connexions à un serveur SSL incorrect avec un nom distinctif qui ne correspond pas.

   Définissez les propriétés **SslTrustStore** et **SslKeyStore** sur des noms de fichier qui pointent vers les fichiers de clés certifiées et les fichiers de clés. Si vous configurez ces propriétés pour un agent déjà en cours d'exécution, arrêtez et redémarrez l'agent pour qu'il se reconnecte en mode SSL.

   Les fichiers de propriétés contiennent des mots de passe en texte en clair. Il est donc judicieux de définir les droits d'accès appropriés au système de fichiers.

   Pour plus d'informations sur les propriétés SSL, voir Propriétés SSL pour MFT.
5. Si un gestionnaire de files d'attente d'agent utilise SSL, vous ne pouvez pas fournir les détails nécessaires lors de la création de l'agent. Procédez comme suit pour créer l'agent:
   a) Créez l'agent à l'aide de la commande **fteCreateAgent** . Vous recevez un avertissement indiquant que vous ne parvenez pas à publier l'existence de l'agent dans le gestionnaire de file d'attente de coordination.
   b) Editez le fichier `agent.properties` créé à l'étape précédente pour ajouter les informations SSL. Lorsque l'agent est correctement démarré, une nouvelle tentative de publication est effectuée.
6. Si des agents ou des instances de IBM MQ Explorer sont en cours d'exécution alors que les propriétés SSL du fichier `agent.properties` ou du fichier `coordination.properties` sont modifiées, vous devez redémarrer l'agent ou IBM MQ Explorer.

Référence associée

Le fichier MFT `agent.properties`

Connexion à un gestionnaire de files d'attente en mode client avec authentification de canal

IBM WebSphere MQ 7.1 a introduit des enregistrements d'authentification de canal pour contrôler plus précisément l'accès au niveau du canal. Ce changement de comportement signifie que par défaut, les gestionnaires de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure nouvellement créés rejettent les connexions client à partir du composant Managed File Transfer .

Pour plus d'informations sur l'authentification de canal, voir «Enregistrements d'authentification de canal», à la page 50.

Si la configuration de l'authentification de canal pour le SVRCONN utilisé par Managed File Transfer spécifie un ID MCAUSER non privilégié, vous devez accorder des enregistrements de droits d'accès spécifiques pour le gestionnaire de files d'attente, les files d'attente et les rubriques, afin de permettre au Managed File Transfer Agent et aux commandes de fonctionner correctement. Utilisez la commande MQSC SET CHLAUTH ou la commande PCF Set Channel Authentication Record pour créer, modifier ou supprimer des enregistrements d'authentification de canal. Pour tous les agents Managed File Transfer que vous souhaitez connecter au gestionnaire de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure, vous pouvez soit configurer un ID MCAUSER à utiliser pour tous vos agents, soit configurer un ID MCAUSER distinct pour chaque agent.

Accordez à chaque ID MCAUSER les droits suivants:

- Enregistrements de droits d'accès requis pour le gestionnaire de files d'attente:
  - connect
  - setid
  - inq
- Enregistrements de droits d'accès requis pour les files d'attente.

  Pour toutes les files d'attente spécifiques à l'agent, c'est-à-dire les noms de file d'attente qui se terminent par *nom_agent* dans la liste suivante, vous devez créer ces enregistrements de droits d'accès aux files d'attente pour chaque agent que vous souhaitez connecter au gestionnaire de files d'attente IBM WebSphere MQ 7.1 ou version ultérieure à l'aide d'une connexion client.
  - put, get, dsp (SYSTEM.DEFAULT.MODEL.QUEUE)
  - put, get, setid, browse (SYSTEM.FTE.COMMAND.*nom_agent*)
  - put, get (SYSTEM.FTE.DATA.*nom_agent*)
  - put, get (SYSTEM.FTE.REPLY.*nom_agent*)
  - put, get, inq, browse (SYSTEM.FTE.STATE.*nom_agent*)
  - put, get, browse (SYSTEM.FTE.EVENT.*nom_agent*)
  - put, get (SYSTEM.FTE)
- Enregistrements de droits d'accès requis pour les rubriques:
  - sub, pub (SYSTEM.FTE)
- Enregistrements de droits d'accès requis pour les transferts de fichiers.

  Si vous disposez d'ID MCAUSER distincts pour l'agent source et l'agent de destination, créez les enregistrements de droits d'accès dans les files d'attente des agents à la fois à la source et à la destination.

  Par exemple, si l'ID MCAUSER de l'agent source est **user1** et l'ID MCAUSER de l'agent cible est **user2**, définissez les droits suivants pour les utilisateurs de l'agent:

Utilisateur d'agent	File d'attente	droits requis
user1	SYSTEME SYSTEM.FTE.DATA.*nom_agent_destination*	put
user1	SYSTEME SYSTEM.FTE.COMMAND.*nom_agent_destination*	put
user2	SYSTEME SYSTEM.FTE.REPLY.*nom_agent_source*	put
user2	SYSTEME SYSTEM.FTE.COMMAND.*nom_agent_source*	put

Configuration de SSL ou TLS entre l'agent de pont Connect:Direct et le noeud Connect:Direct

Configurez l'agent de pont Connect:Direct et le noeud Connect:Direct pour qu'ils se connectent via le protocole SSL en créant un magasin de clés et un magasin de clés de confiance et en définissant les propriétés dans le fichier de propriétés de l'agent de pont Connect:Direct .

Pourquoi et quand exécuter cette tâche

Ces étapes incluent des instructions permettant d'obtenir vos clés signées par une autorité de certification. Si vous n'utilisez pas d'autorité de certification, vous pouvez générer un certificat autosigné. Pour plus d'informations sur la génération d'un certificat autosigné, voir «Utilisation de SSL/TLS sous UNIX, Linux, and Windows», à la page 294.

Ces étapes incluent des instructions pour la création d'un magasin de clés et d'un magasin de clés de confiance pour l'agent de pont Connect:Direct . Si l'agent de pont Connect:Direct possède déjà un

magasin de clés et un magasin de clés de confiance qu'il utilise pour se connecter de manière sécurisée aux gestionnaires de files d'attente IBM MQ , vous pouvez utiliser le magasin de clés et le magasin de clés de confiance existants lors de la connexion sécurisée au noeud Connect:Direct . Pour plus d'informations, voir «Configuration du chiffrement SSL ou TLS pour MFT», à la page 571.

Procédure

Pour le noeud Connect:Direct , procédez comme suit:

1. Générez une clé et un certificat signé pour le noeud Connect:Direct .

   Pour ce faire, utilisez l'outil IBM Key Management fourni avec IBM MQ. Pour plus d'informations, voir «Utilisation de SSL/TLS», à la page 282.
2. Envoyez une demande à une autorité de certification pour que la clé soit signée. Vous recevez un certificat en retour.
3. Créez un fichier texte, par exemple `/test/ssl/certs/CAcert`, qui contient la clé publique de votre autorité de certification.
4. Installez l'option Secure + sur le noeud Connect:Direct .

   Si le noeud existe déjà, vous pouvez installer l'option Secure + en exécutant à nouveau le programme d'installation, en indiquant l'emplacement de l'installation existante et en choisissant d'installer uniquement l'option Secure +.
5. Créez un nouveau fichier texte ; par exemple, `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt`.
6. Copiez le certificat que vous avez reçu de votre autorité de certification et la clé privée, qui se trouve dans `/test/ssl/cd/privateKeys/`*`node_name`*`.key`, dans le fichier texte.

   Le contenu de `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt` doit être au format suivant:

```
-----BEGIN CERTIFICATE-----
MIICnzCCAgigAwIBAgIBGjANBgkqhkiG9w0BAQUFADBeMQswCQYDVQQGEwJHQjES
MBAGA1UECBMJSGFtcHNoaXJlMRAwDgYDVQQHEwdIdXJzbGV5MQwwCgYDVQQKEwNJ
Qk0xDjAMBgNVBAsTBU1RSVBUMQswCQYDVQQDEwJDQTAeFw0xMTAzMDExNjIwNDZa
Fw0yMTAyMjYxNjIwNDZaMFAxCzAJBgNVBAYTAkdCMRIwEAYDVQQIEwlIYW1wc2hp
cmUxDDAKBgNVBAoTA0lCTTEOMAwGA1UECxMFTVFGVEUxDzANBgNVBAMTBmJpbmJh
ZzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAvgP1QIklU9ypSKD1XoODo1yk
EyMFXBOUpZRrDVxjoSEC0vtWNcJ199e+Vc4UpNybDyBu+NkDlMNofX4QxeQcLAFj
WnhakqCiQ+JIAD5Aurhnrwche0MV3kjA84GKH/rOSVqtl984mu/lDyS819XcfSSn
cOOMsK1KbneVSCIV2XECAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0E
HxYdT3BlblNTTCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFNXMIpSc
csBXUniW4A3UrZnCRsv3MB8GA1UdIwQYMBaAFDXY8rmj4lVz5+FVAoQb++cns+B4
MA0GCSqGSIb3DQEBBQUAA4GBAFc7klXa4pGKYgwchxKpE3ZF6FNwy4vBXS216/ja
8h/vl8+iv01OCL8t0ZOKSU95fyZLzOPKnCH7v+ItFSE3CIiEk9Dlz2U6WO9lICwn
l7PL72TdfaL3kabwHYVf17IVcuL+VZsZ3HjLggP2qHO9ZuJPspeT9+AxFVMLiaAb
8eHw
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,64A02DA15B6B6EF9

57kqxLOJ/gRUOIQ6hVK2YN13B4E1jAi1gSme0I5ZpEIG8CHXISKB7/0cke2FTqsV
lvI99QyCxsDWoMNt5fj51v7aPmVeS60bOm+UlGre8B/Ze18JVj2O4K2Uh72rDCXE
5e6eFxSdUM207sQDy20euBVELJtM2kOkL1ROdoQQSlU3XQNgJw/t3ZIx5hPXWEQT
rjRQO64BEhb+PzzxPF8uwzZ9IrUK9BJ/UUnqC6OdBR87IeA4pnJD1Jvb2ML7EN9Z
5Y+50hTKI8OGvBvWXO4fHyvIX5aslwhBoArXIS1AtNTrptPvoaP1zyIAeZ6OCVo/
SFo+A2UhmtEJeOJaZG2XZ3H495fAw/EHmjehzIACwukQ9nSIETgu4A1+CV64RJED
aYBCM8UjaAkbZDH5gn7+eBov0ssXAXWDyJBVhUOjXjvAj/e1h+kcSF1hax5D//AI
66nRMZzboSxNqkjcVd8wfDwP+bEjDzUaaarJTS7lIFeLLw7eJ8MNAkMGicDkycL0
EPBU9X5QnHKLKOfYHN/1WgUk8qt3UytFXXfzTXGF3EbsWbBupkT5e5+lYcX8OVZ6
sHFPNlHluCNy/riUcBy9iviVeodX8IomOchSyO5DKl8bwZNjYtUP+CtYHNFU5BaD
I+1uUOAeJ+wjQYKT1WaeIGZ3VxuNITJul8y5qDTXXfX7vxM5OoWXa6U5+AYuGUMg
/itPZmUmNrHjTk7ghT6i1IQOaBowXXKJBlMmq/6BQXN2IhkD9ys2qrvM1hdi5nAf
egmdiG50loLnBRqWbfR+DykpAhK4SaDi2F52Uxovw3Lhiw8dQP7lzQ==
-----END RSA PRIVATE KEY-----
```

7. Démarrez l'outil d'administration Secure +.
   - Sur les systèmes Linux ou UNIX , exécutez la commande **spadmin.sh**.
   - Sur les systèmes Windows , cliquez sur **Démarrer** > **Programmes** > **Sterling Commerce Connect:Direct** > **CD Secure + Admin Tool**

L'outil d'administration CD Secure + démarre.

8. Dans l'outil d'administration CD Secure +, cliquez deux fois sur **.Ligne** locale pour éditer les paramètres SSL ou TLS principaux.
   a) Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
   b) Sélectionnez **Désactiver la substitution**.
   c) Sélectionnez au moins une suite de chiffrement.
   d) Si vous souhaitez une authentification bidirectionnelle, remplacez la valeur de **Activer l'authentification client** par `Yes`.
   e) Dans la zone **Certificat racine accrédité** , entrez le chemin d'accès au fichier de certificat public de votre autorité de certification, `/test/ssl/certs/CAcert`.
   f) Dans la zone **Fichier de certificat de clé** , entrez le chemin d'accès au fichier que vous avez créé, `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt`.
9. Cliquez deux fois sur le **.Ligne** du client pour éditer les paramètres SSL ou TLS principaux.
   a) Sélectionnez **Activer le protocole SSL** ou **Activer le protocole TLS**, selon le protocole que vous utilisez.
   b) Sélectionnez **Désactiver la substitution**.

Pour l'agent de pont Connect:Direct , procédez comme suit:

10. Créez un magasin de clés de confiance. Pour ce faire, vous pouvez créer une clé factice, puis la supprimer.

    Vous pouvez utiliser les commandes suivantes:

    ```
    keytool -genkey -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
    ```

    ```
    keytool -delete -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
    ```

11. Importez le certificat public de l'autorité de certification dans le magasin de clés de confiance.

    Vous pouvez utiliser la commande suivante :

    ```
    keytool -import -trustcacerts -alias myCA
            -file /test/ssl/certs/CAcert
            -keystore /test/ssl/fte/stores/truststore.jks
    ```

12. Editez le fichier de propriétés de l'agent de pont Connect:Direct .

    Incluez les lignes suivantes n'importe où dans le fichier:

    ```
    cdNodeProtocol=protocol
    cdNodeTruststore=/test/ssl/fte/stores/truststore.jks
    cdNodeTruststorePassword=password
    ```

    Dans l'exemple de cette étape, *protocol* est le protocole que vous utilisez, SSL ou TLS, et *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés de confiance.

13. Si vous souhaitez une authentification bidirectionnelle, créez une clé et un certificat pour l'agent de pont Connect:Direct .
    a) Créez un magasin de clés et une clé.

       Vous pouvez utiliser la commande suivante :

       ```
       keytool -genkey -keyalg RSA -alias agent_name
               -keystore /test/ssl/fte/stores/keystore.jks
               -storepass password -validity 365
       ```

    b) Générez une demande de signature.

Vous pouvez utiliser la commande suivante :

```
keytool -certreq -v -alias agent_name
        -keystore /test/ssl/fte/stores/keystore.jks -storepass password
        -file /test/ssl/fte/requests/agent_name.request
```

c) Importez le certificat que vous avez reçu à l'étape précédente dans le magasin de clés. Le certificat doit être au format x.509 .

Vous pouvez utiliser la commande suivante :

```
keytool -import -keystore /test/ssl/fte/stores/keystore.jks
        -storepass password -file certificate_file_path
```

d) Editez le fichier de propriétés de l'agent de pont Connect:Direct .

Incluez les lignes suivantes n'importe où dans le fichier:

```
cdNodeKeystore=/test/ssl/fte/stores/keystore.jks
cdNodeKeystorePassword=password
```

Dans l'exemple de cette étape, *password* est le mot de passe que vous avez spécifié lors de la création du magasin de clés.

Tâches associées

Configuration du pont Connect:Direct

ULW Sécurisation des clients AMQP

Vous utilisez une série de mécanismes de sécurité pour sécuriser les connexions des clients AMQP et vous assurer que les données sont correctement protégées sur le réseau. Vous pouvez générer la sécurité dans vos applications MQ Light . Vous pouvez également utiliser les fonctions de sécurité existantes d' IBM MQ avec les clients AMQP, de la même manière que les fonctions sont utilisées pour d'autres applications.

Règles d'authentification de canal (CHLAUTH)

Vous pouvez utiliser des règles d'authentification de canal afin de restreindre les connexions TCP à un gestionnaire de files d'attente. Les canaux AMQP prennent en charge l'utilisation de règles d'authentification de canal que vous configurez pour votre gestionnaire de files d'attente. Si des règles d'authentification de canal sont définies avec un profil qui correspond à des canaux AMQP dans votre gestionnaire de files d'attente, elles sont appliquées à ces canaux. Par défaut, l'authentification de canal est activée dans les nouveaux gestionnaires de files d'attente IBM MQ ; par conséquent, vous devez effectuer au moins quelques étapes de configuration avant de pouvoir utiliser un canal AMQP.

Pour plus d'informations sur la configuration des règles d'authentification de canal pour autoriser les connexions AMQP à votre gestionnaire de files d'attente, voir Création et utilisation de canaux AMQP.

Authentification de connexion (CONNAUTH)

Vous pouvez utiliser l'authentification de connexion pour authentifier les connexions à un gestionnaire de files d'attente. Les canaux AMQP prennent en charge l'utilisation de l'authentification de connexion pour contrôler l'accès au gestionnaire de files d'attente depuis des applications AMQP.

Le protocole AMQP utilise l'infrastructure SASL (Simple Authentication and Security Layer) pour spécifier la façon dont une connexion est authentifiée. Il existe divers mécanismes SASL et IBM MQ en prend en charge deux : ANONYMOUS et PLAIN.

Dans le cas de ANONYMOUS, aucune donnée d'identification n'est transmise du client au gestionnaire de files d'attente pour l'authentification. Si l'objet MQ AUTHINFO spécifié dans l'attribut CONNAUTH possède

la valeur REQUIRED ou REQDADM (en cas de connexion en tant qu'administrateur) pour CHCKCLNT, la connexion est refusée. Si la valeur de CHCKCLNT est NONE ou OPTIONAL, la connexion est acceptée.

Dans le cas de PLAIN, un nom d'utilisateur et un mot de passe sont transmis du client au gestionnaire de files d'attente pour l'authentification. Si l'objet MQ AUTHINFO spécifié dans l'attribut CONNAUTH possède la valeur NONE pour CHCKCLNT, la connexion est refusée. Si la valeur de CHCKCLNT est OPTIONAL, REQUIRED ou REQDADM (en cas de connexion en tant qu'administrateur), le nom d'utilisateur et le mot de passe sont vérifiés par le gestionnaire de files d'attente. Ce dernier vérifie le système d'exploitation (si l'objet AUTHINFO est de type IDPWOS) ou un référentiel LDAP (si l'objet AUTHINFO est de type IDPWLDAP).

Le tableau suivant présente un récapitulatif de ce comportement d'authentification :

Tableau 95. Récapitulatif des mécanismes SASL et de l'authentification de connexion

Mécanisme SASL	**Données d'identification transmises du client au gestionnaire de files d'attente ?**	**Valeur CHKCLNT**
ANONYMOUS	Non	REQUIRED ou REQDADM - connexion refusée NONE ou OPTIONAL - connexion acceptée
PLAIN	Oui, nom d'utilisateur et mot de passe	REQUIRED, REQDADM ou OPTIONAL - nom d'utilisateur et mot de passe vérifiés par le gestionnaire de files d'attente NONE - connexion refusée

Si vous utilisez un client MQ Light, vous pouvez spécifier des données d'identification en les incluant dans l'adresse AMQP à laquelle vous vous connectez, par exemple :

```
amqp://mwhitehead:mYp4ssw0rd@localhost:5672/sports/football
```

Paramètre MCAUSER sur un canal

Les canaux AMQP possèdent un attribut MCAUSER que vous pouvez utiliser pour définir l'ID utilisateur IBM MQ sous lequel toutes les connexions à un canal particulier sont autorisées. Toutes les connexions depuis des clients AMQP à ce canal adoptent l'ID MCAUSER que vous avez configuré. Cet ID utilisateur est employé pour l'autorisation de la messagerie dans différentes rubriques.

Il est recommandé d'utiliser l'authentification de canal (CHLAUTH) pour sécuriser les connexions aux gestionnaires de files d'attente. Si vous utilisez l'authentification de canal, il est recommandé de configurer un utilisateur non privilégié comme valeur pour MCAUSER. Ainsi, si une connexion à un canal n'est pas mise en correspondance par une règle CHLAUTH, elle ne sera pas autorisée à effectuer des opérations de messagerie dans le gestionnaire de files d'attente.

Remarque : Windows Sous Windows, avant IBM MQ 9.1.1, le paramètre d'ID utilisateur MCAUSER n'est pris en charge que pour les ID utilisateur comportant 12 caractères ou moins.

V 9.1.1 Depuis IBM MQ 9.1.1, la limite de 12 caractères n'existe plus.

prise en charge de SSL/TLS

Les canaux AMQP prennent en charge le chiffrement SSL/TLS à l'aide de clés provenant du référentiel de clés configuré pour votre gestionnaire de files d'attente. Les options de configuration de canal AMQP pour le chiffrement SSL/TLS prennent en charge les mêmes options que les autres types de canal MQ ; vous pouvez préciser une spécification de chiffrement et indiquer si le gestionnaire de files d'attente requiert des certificats des connexions client AMQP.

A l'aide des attributs FIPS du gestionnaire de files d'attente, vous pouvez contrôler les suites de chiffrement SSL/TLS que vous pouvez utiliser pour sécuriser les connexions depuis les clients AMQP.

Pour plus d'informations sur la configuration d'un référentiel de clés pour le gestionnaire de files d'attente, voir Utilisation de SSL ou TLS sur les systèmes UNIX, Linux et Windows.

Pour plus d'informations sur la configuration de la prise en charge de SSL/TLS pour une connexion client AMQP, voir Création et utilisation de canaux AMQP.

service JAAS

Si vous le souhaitez, vous pouvez configurer des canaux AMQP avec un module de connexion JAAS qui peut vérifier le nom d'utilisateur et le mot de passe indiqués par un client AMQP. Voir «Configuration de JAAS pour les canaux AMQP», à la page 579.

Tâches associées

Développement d'applications client AMQP

Création et utilisation de canaux AMQP

ULW Restriction de la reprise du client AMQP

Lorsqu'une connexion client AMQP dont l'identificateur de client est identique à celui d'une connexion client AMQP existante est établie, la connexion client existante est déconnectée par défaut. Toutefois, vous pouvez configurer le gestionnaire de files d'attente afin de restreindre le comportement de reprise de client pour que la reprise ne soit possible que lorsque certains critères sont satisfaits.

Par exemple, la déconnexion de la connexion client existante peut ne pas être appropriée si des applications AMQP sont développées par différentes équipes et qu'elles utilisent le même ID de client. Pour résoudre ce problème, vous pouvez restreindre la reprise de client reposant sur le nom du canal AMQP utilisé, l'adresse IP du client et l'ID utilisateur du client (lorsque l'authentification SASL est activée).

Utilisez les paramètres des attributs de gestionnaire de files d'attente **AdoptNewMCA** et **AdoptNewMCACheck** pour spécifier le niveau requis de la restriction de reprise du client, comme indiqué dans le tableau suivant:

*Tableau 96. Paramètres **AdoptNewMCA** et **AdoptNewMCACheck** pour restreindre la reprise du client*

AdoptNewMCA	**AdoptNewMCACheck**	**Critères vérifiés avant l'autorisation de la reprise de client**
NO ou indéfini	Non applicable	Aucune. La reprise de client est autorisée pour toutes les connexions client qui sont authentifiées et qui satisfont toutes les règles CHLAUTH.
ALL (ou une valeur autre que NO)	QM ou indéfini	Aucune. La reprise de client est autorisée pour toutes les connexions client qui sont authentifiées et qui satisfont toutes les règles CHLAUTH.

*Tableau 96. Paramètres **AdoptNewMCA** et **AdoptNewMCACheck** pour restreindre la reprise du client (suite)*

AdoptNewMCA	AdoptNewMCACheck	Critères vérifiés avant l'autorisation de la reprise de client
ALL (ou une valeur autre que NO)	NOM	ID utilisateur (lorsque SASL est activé) Nom du canal
ALL (ou une valeur autre que NO)	ADDRESS	ID utilisateur (lorsque SASL est activé) Adresse IP
ALL (ou une valeur autre que NO)	TOUT	ID utilisateur (lorsque SASL est activé) Nom du canal Adresse IP

Les attributs de gestionnaire de files d'attente **AdoptNewMCA** et **AdoptNewMCACheck** font partie de la configuration de gestionnaire de files d'attente, qui est définie dans la strophe CHANNELS. Sur les systèmes IBM MQ for Windows et IBM MQ for Linux x86-64 , modifiez les informations de configuration à l'aide du IBM MQ Explorer. Sur les autres systèmes, modifiez les informations en éditant le fichier de configuration `qm.ini`. Pour des informations sur la modification des informations relatives aux canaux de gestionnaire de files d'attente, voir la rubrique relative aux attributs des canaux.

Tâches associées

Développement d'applications client AMQP

Création et utilisation de canaux AMQP

ULW

Configuration de JAAS pour les canaux AMQP

Les modules personnalisés JAAS (Java Authentication and Authorization Service) peuvent être utilisés pour authentifier les données d'identification composées d'un nom d'utilisateur et d'un mot de passe qui sont transmises à un canal AMQP par un client AMQP lorsqu'il se connecte.

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser un module JAAS personnalisé si vous utilisez déjà des modules JAAS pour l'authentification dans d'autres systèmes reposant sur Java et voulez réutiliser ces modules pour l'authentification des connexions AMQP à MQ. Vous pouvez aussi écrire un modèle JAAS personnalisé si les fonctions d'authentification intégrées à MQ ne prennent pas en charge le mécanisme d'authentification que vous utilisez.

La configuration de modules JAAS pour des canaux AMQP est effectuée au niveau du gestionnaire de files d'attente. Cela signifie que si vous configurez un module JAAS pour l'authentification des connexions AMQP au gestionnaire de files d'attente, le module est appliqué à tous les canaux AMQP. Le nom du canal qui a appelé le module JAAS est transmis au module, ce qui vous permet de coder des comportements de connexion JAAS différents pour des canaux différents.

D'autres informations sont également transmises au module JAAS :

- L'ID du client AMQP qui tente de s'authentifier.
- L'adresse réseau du client AMQP.
- Le nom du canal qui a appelé le module JAAS.

Procédure

Pour configurer un module de configuration JAAS pour les canaux AMQP, procédez comme suit :

1. Définissez un fichier `jaas.config` contenant une ou plusieurs strophes de configuration de module JAAS. La section doit spécifier le nom qualifié complet de la classe Java qui implémente l'interface JAAS `javax.security.auth.spi.LoginModule` .
   - Un fichier `jaas.config` par défaut est fourni avec le produit et se trouve dans *QM_data_directory*`/amqp/jaas.config`.
   - Une strophe préconfigurée nommée `MQXRConfig` est déjà définie dans le fichier `jaas.config` par défaut.
2. Spécifiez le nom de la strophe à utiliser pour les canaux AMQP.
   - 

Ajoutez une propriété au fichier `amqp_unix.properties`.
   - 

Ajoutez une propriété au fichier `amqp_win.properties`.

   Le format de la propriété est le suivant :

   ```
   com.ibm.mq.MQXR.JAASConfig=JAAS_stanza_name
   ```

   Par exemple :

   ```
   com.ibm.mq.MQXR.JAASConfig=MQXRConfig
   ```
3. Configurez l'environnement de gestionnaire de files d'attente afin d'inclure la classe du module personnalisé. Le service AMQP doit avoir accès à la classe Java configurée dans la strophe de configuration de JAAS.

   Pour ce faire, ajoutez le chemin d'accès à la classe JAAS dans le fichier `service.env` de MQ. Editez le fichier `service.env` dans le répertoire de configuration MQ (*MQ_config_directory*) ou dans le répertoire de configuration du gestionnaire de files d'attente (*QM_config_directory*) pour définir la variable CLASSPATH sur l'emplacement de la classe de module JAAS .

Que faire ensuite

Un exemple de module de connexion JAAS est fourni avec le produit dans le répertoire *mq_installation_directory*`/amqp/samples` . Il authentifie toutes les connexions client, quel que soit le nom d'utilisateur ou le mot de passe avec lequel le client se connecte.

Vous pouvez modifier le code source de l'exemple et le recompiler pour tenter de n'authentifier que des utilisateurs spécifiques qui se servent d'un mot de passe particulier. Pour configurer le canal AMQP sur un système UNIX afin d'utiliser l'exemple de module de connexion JAAS livré avec le produit :

1. Editez le fichier `/var/mqm/qmgrs/QMNAME/amqp/amqp_unix.properties` et définissez la propriété `com.ibm.mq.MQXR.JAASConfig=MQXRConfig`.
2. Editez le fichier /var/mqm/service.env et définissez la propriété CLASSPATH=*mq_installation_location*`/amqp/samples`

Le fichier `jaas.config` contient déjà une strophe nommée `MQXRConfig` qui spécifie l'exemple de classe `samples.JAASLoginModule` comme classe de module de connexion. Il n'est pas nécessaire de modifier le fichier `jaas.config` avant d'utiliser l'exemple de module.

Tâches associées

Développement d'applications client AMQP

Création et utilisation de canaux AMQP

Advanced Message Security

Advanced Message Security (AMS) est un composant de IBM MQ qui offre un niveau de protection élevé pour les données sensibles transitant par le réseau IBM MQ , sans affecter les applications finales.

Présentation des Advanced Message Security

Les applications IBM MQ peuvent utiliser Advanced Message Security pour envoyer des données sensibles, telles que des transactions financières à valeur élevée et des informations personnelles, avec différents niveaux de protection à l'aide d'un modèle de cryptographie à clé publique.

Référence associée

Codes retour GSKit utilisés dans les messages AMS

Caractéristiques et fonctions de Advanced Message Security

Advanced Message Security développe les services de sécurité IBM MQ pour fournir la signature et le chiffrement des données au niveau des messages. Les services étendus garantissent que les données de message n'ont pas été modifiées entre le moment où elles sont placées à l'origine dans une file d'attente et le moment où elles sont extraites. En outre, AMS vérifie qu'un expéditeur de données de message est autorisé à placer des messages signés dans une file d'attente cible.

AMS fournit les fonctions suivantes:

- Sécurise les transactions sensibles ou à valeur élevée traitées par IBM MQ.
- Détecte et supprime les messages malveillants ou non autorisés avant qu'ils ne soient traités par une application de réception.
- Vérifie que les messages n'ont pas été modifiés lors de leur passage de la file d'attente à la file d'attente.
- Protège les données non seulement lorsqu'elles circulent sur le réseau, mais également lorsqu'elles sont placées dans une file d'attente.
- Sécurise les applications propriétaires et écrites par le client existantes pour IBM MQ.
- V 9.1.3 z/OS Depuis la IBM MQ 9.1.3, IBM MQ for z/OS offre la possibilité de supprimer et d'ajouter éventuellement une protection AMS à partir ou vers des messages qui transitent sur le réseau, respectivement. Il s'agit de l'exception *Server to Server Message Channel Agent (MCA) Interception.*.
- ULW V 9.1.0.4 V 9.1.4 Depuis IBM MQ 9.1.4 et IBM MQ 9.1.0 Fix Pack 4, une vérification a été ajoutée au code de la bibliothèque IBM MQ qui s'exécute dans le programme d'application du client. Celle-ci s'exécute au début de son initialisation pour lire la valeur de la variable d'environnement *AMQ_AMS_FIPS_OFF*. Si cette variable d'environnement est associée à une valeur, le code GSKit est alors exécuté en mode non FIPS dans cette application.

Qualités de protection disponibles avec AMS

Il existe trois qualités de protection pour Advanced Message Security, `Integrity`, `Privacy`et `Confidentiality`.

La protection `Integrity` est assurée par la signature numérique, qui permet de savoir qui a créé le message et que le message n'a pas été modifié ou altéré.

La protection `Privacy` est assurée par une combinaison de signature numérique et de chiffrement. Le chiffrement garantit que les données de message ne sont visibles que par le ou les destinataires prévus. Même si des destinataires non autorisés obtiennent une copie des données de message chiffrées, ils ne peuvent pas afficher eux-mêmes les données de message réelles.

La protection `Confidentiality` est fournie par le chiffrement uniquement avec une réutilisation de clé facultative.

Effet sur les performances

AMS utilise une combinaison de routines cryptographiques symétriques et asymétriques pour fournir la signature numérique et le chiffrement. Etant donné que les opérations de clé symétrique sont très rapides par rapport aux opérations de clé asymétrique, qui consomment beaucoup d'UC, cela peut avoir un impact significatif sur les coûts de protection d'un grand nombre de messages avec AMS.

Routines cryptographiques asymétriques
Par exemple, lors de l'insertion d'un message signé, le hachage de message est signé à l'aide d'une opération de clé asymétrique.

Lors de l'obtention d'un message signé, une autre opération de clé asymétrique est utilisée pour vérifier le hachage signé.

Par conséquent, un minimum de deux opérations de clé asymétrique est requis par message pour signer et vérifier les données de message.

Routines cryptographiques asymétriques et symétriques
Lors de l'insertion d'un message chiffré, une clé symétrique est générée, puis chiffrée à l'aide d'une opération de clé asymétrique pour chaque destinataire prévu du message.

Les données de message sont ensuite chiffrées avec la clé symétrique. Lors de l'obtention du message chiffré, le destinataire prévu doit utiliser une opération de clé asymétrique pour reconnaître la clé symétrique utilisée pour le message.

Par conséquent, les trois qualités de protection contiennent des éléments différents des opérations de clés asymétriques à forte intensité d'UC, ce qui aura un impact significatif sur le débit de messagerie maximal réalisable pour les applications qui envoient et reçoivent des messages.

Toutefois, les règles `Confidentiality` permettent la réutilisation de clés symétriques sur une séquence de messages. Grâce à la réutilisation de clé symétrique, les politiques `Confidentiality` permettent des économies de coût importantes de l'UC. Ce mode de fonctionnement continue d'utiliser le format PKCS#7 pour partager une clé de chiffrement symétrique. Toutefois, il n'existe pas de signature numérique, ce qui élimine certaines des opérations de clé asymétrique par message. La clé symétrique doit quand même être chiffrée avec des opérations de clé asymétrique pour chaque destinataire, mais la clé symétrique peut éventuellement être réutilisée dans plusieurs messages destinés aux mêmes destinataires. Si la réutilisation de clé est autorisée par la politique, seul le premier message requiert des opérations de clé asymétrique. Les messages suivants doivent utiliser uniquement des opérations de clé symétrique.

Réutilisation de clé

Avec les règles `Confidentiality` , vous pouvez utiliser l'approche de réutilisation de clé symétrique pour réduire de manière significative les coûts liés au chiffrement d'un certain nombre de messages placés dans la même file d'attente et destinés au ou aux mêmes destinataires.

Par exemple, lors de l'insertion de 10 messages chiffrés dans le même ensemble de destinataires, une clé symétrique est générée, puis chiffrée pour le premier message, à l'aide d'une opération de clé asymétrique pour chaque destinataire prévu du message.

En fonction des limites contrôlées par la règle, la clé symétrique chiffrée peut ensuite être réutilisée par les messages ultérieurs destinés aux mêmes destinataires. Une application qui obtient des messages chiffrés peut appliquer la même optimisation, dans la mesure où l'application peut détecter lorsqu'une clé symétrique n'a pas été modifiée et éviter les frais liés à l'extraction de la clé symétrique.

Dans cet exemple, 90% des opérations de clé asymétrique peuvent être évitées par les applications d'insertion et d'obtention en réutilisant la même clé.

Pour plus d'informations sur l'utilisation de la réutilisation des clés, voir:

- Commande MQSC SET POLICY
- Commande de contrôle setmqspl
- IBM i IBM i commande SETMQMSPL

Concepts clés dans AMS

Découvrez les concepts clés de Advanced Message Security pour comprendre comment l'outil fonctionne et comment le gérer efficacement.

Infrastructure à clés publiques et Advanced Message Security

L'infrastructure à clé publique (ICP) est un système d'installations, de politiques et de services qui appuient l'utilisation de la cryptographie à clé publique pour obtenir des communications sécurisées.

Il n'existe pas de norme unique qui définit les composants d'une infrastructure à clé publique, mais une ICP implique généralement l'utilisation de certificats de clé publique et comprend des autorités de certification (CA) et d'autres autorités d'enregistrement (RA) qui fournissent les services suivants:

- Emission de certificats numériques
- Validation des certificats numériques
- Révocation de certificats numériques
- Distribution de certificats

L'identité des utilisateurs et des applications est représentée par la zone **Nom distinctif (DN)** dans un certificat associé à des messages signés ou chiffrés. Advanced Message Security utilise cette identité pour représenter un utilisateur ou une application. Pour authentifier cette identité, l'utilisateur ou l'application doit avoir accès au magasin de clés dans lequel le certificat et la clé privée associée sont stockés. Chaque certificat est représenté par un libellé dans le magasin de clés.

Concepts associés

«Utilisation de magasins de clés et de certificats», à la page 626
Pour fournir une protection cryptographique transparente aux applications IBM MQ , Advanced Message Security utilise le fichier de clés, dans lequel sont stockés les certificats de clé publique et une clé privée. Sous z/OS, un fichier de clés SAF est utilisé à la place d'un fichier de clés.

Certificats numériques dans AMS

Advanced Message Security associe les utilisateurs et les applications à des certificats numériques standard X.509 . Les certificats X.509 sont généralement signés par une autorité de certification de confiance et impliquent des clés privées et publiques qui sont utilisées pour le chiffrement et le déchiffrement.

Les certificats numériques offrent une protection contre l'usurpation d'identité en liant une clé publique à son propriétaire, qu'il s'agisse d'un individu, d'un gestionnaire de files d'attente ou d'une autre entité. Les certificats numériques sont également appelés certificats de clé publique, car ils vous donnent l'assurance de la propriété d'une clé publique lorsque vous utilisez un schéma de clé asymétrique. Ce schéma requiert la génération d'une clé publique et d'une clé privée pour une application. Les données chiffrées à l'aide de la clé publique ne peuvent être déchiffrées qu'à l'aide de la clé privée correspondante, tandis que les données chiffrées à l'aide de la clé privée ne peuvent être déchiffrées qu'à l'aide de la clé publique correspondante. La clé privée est stockée dans un fichier de base de données de clés protégé par mot de passe. Seul son propriétaire a accès à la clé privée utilisée pour déchiffrer les messages qui sont chiffrés à l'aide de la clé publique correspondante.

Si les clés publiques sont envoyées directement par leur propriétaire à une autre entité, il existe un risque que le message soit intercepté et que la clé publique soit remplacée par une autre. C'est ce qu'on appelle une attaque de type "man-in-the-middle". La solution consiste à échanger des clés publiques par l'intermédiaire d'un tiers de confiance, ce qui permet à l'utilisateur de s'assurer que la clé publique appartient à l'entité avec laquelle vous communiquez. Au lieu d'envoyer votre clé publique directement, vous demandez à un tiers de confiance de l'incorporer dans un certificat numérique. Le tiers de confiance qui émet des certificats numériques est appelé une autorité de certification (CA).

Pour plus d'informations sur les certificats numériques, voir Qu'est-ce qu'un certificat numérique?.

Un certificat numérique contient la clé publique d'une entité et indique que la clé publique appartient à cette entité:

- lorsqu'un certificat est destiné à une entité individuelle, il est appelé *certificat personnel* ou *certificat d'utilisateur*.
- lorsqu'un certificat est destiné à une autorité de certification, le certificat est appelé *certificat de l'autorité de certification* ou *certificat de signataire*.

Remarque : Advanced Message Security prend en charge les certificats autosignés dans les applications Java et natives

Concepts associés

«Cryptographie», à la page 7
La cryptographie est le processus de conversion entre du texte lisible, appelé *texte en clair*, et un format illisible, appelé *texte chiffré*.

Multi *gestionnaire des droits d'accès aux objets*

Sur Multiplatforms, Object Authority Manager (OAM) est le composant de service d'autorisation fourni avec les produits IBM MQ .

L'accès aux entités Advanced Message Security est contrôlé par les groupes d'utilisateurs IBM MQ et la méthode d'accès aux objets (OAM). Les administrateurs peuvent utiliser l'interface de ligne de commande pour accorder ou révoquer des autorisations selon les besoins. Différents groupes d'utilisateurs peuvent avoir différents types de droits d'accès aux mêmes objets. Par exemple, un groupe peut effectuer à la fois des opérations PUT et GET pour une file d'attente spécifique, tandis qu'un autre groupe peut être autorisé uniquement à parcourir la file d'attente. De même, certains groupes peuvent disposer des droits GET et PUT sur une file d'attente, mais ils ne sont pas autorisés à modifier ou à supprimer la file d'attente.

Grâce à la méthode d'accès aux objets (OAM), vous pouvez contrôler:

- Accès aux objets Advanced Message Security via l'interface MQI (Message Queue Interface). Lorsqu'un programme d'application tente d'accéder à des objets, la méthode d'accès aux objets (OAM) vérifie si le profil utilisateur à l'origine de la demande possède l'autorisation pour l'opération demandée. Cela signifie que les files d'attente et les messages des files d'attente peuvent être protégés contre tout accès non autorisé.
- Droit d'utilisation des commandes PCF et MQSC.

Concepts associés

gestionnaire des droits d'accès aux objets

Présentation de l'interface de file d'attente de messages

Technologie prise en charge par Advanced Message Security

Advanced Message Security dépend de plusieurs composants technologiques pour fournir une infrastructure de sécurité.

Advanced Message Security prend en charge les interfaces de programme d'application (API) IBM MQ suivantes:

- interface de file d'attente de messages (MQI)
- IBM MQ Java Message Service (JMS) 1.0.2 et 1.1.
- IBM MQ Classes de base pour Java
- Classes IBM MQ pour. Net en mode non géré

Remarque : Advanced Message Security prend en charge les autorités de certification conformes à X.509 .

Limitations connues de AMS

Un certain nombre d'options IBM MQ ne sont pas prises en charge ou sont soumises à des limitations pour Advanced Message Security.

- Les options IBM MQ suivantes ne sont pas prises en charge ou sont soumises à des limitations:

  Publication/abonnement

  L'un des principaux avantages d'un modèle de messagerie de publication / abonnement sur un point à point est que les applications d'envoi et de réception n'ont pas besoin de se connaître les unes les autres pour que les données soient envoyées et reçues. Cet avantage est annulé par l'utilisation de règles Advanced Message Security qui doivent définir des destinataires ou des signataires autorisés. Il est possible pour une application de publier dans une rubrique via une définition de file d'attente

d'alias qui est protégée par une règle. Il est également possible pour une application d'abonnement d'obtenir des messages à partir d'une file d'attente protégée par une règle. Il n'est pas possible d'affecter une règle directement à une chaîne de rubrique. Les règles ne peuvent être affectées qu'à des définitions de file d'attente.

Conversion de données de canal
Le contenu protégé d'un message protégé Advanced Message Security est transmis au format binaire, ce qui garantit que la conversion des données sur un canal entre les applications n'invalide pas le résumé du message. Les applications qui extraient des messages d'une file d'attente protégée par des règles doivent demander la conversion de données, la conversion du contenu protégé sera tentée une fois les messages vérifiés et non protégés.

Listes de diffusion
Les règles Advanced Message Security peuvent être utilisées lors de la protection des applications qui placent des messages dans des listes de distribution, à condition qu'une règle identique soit définie pour chaque file d'attente de destination de la liste. Si des règles incohérentes sont identifiées lorsqu'une application ouvre une liste de distribution, l'opération d'ouverture échoue et une erreur de sécurité est renvoyée à l'application.

Segmentation des messages d'application
La taille des messages protégés par des règles augmente et les applications ne peuvent pas spécifier avec précision les limites de segment d'un message.

Applications utilisant IBM MQ classes for .NET en mode géré (connexions client)
Les applications utilisant IBM MQ classes for .NET en mode géré (connexions client) ne sont pas prises en charge.

Remarque : L'interception MCA peut être utilisée pour permettre aux clients non pris en charge d'utiliser AMS.

Client Message Service pour les applications .NET (XMS) en mode géré
Les applications du client Message Service pour .NET (XMS) en mode géré ne sont pas prises en charge.

Remarque : L'interception MCA peut être utilisée pour permettre aux clients non pris en charge d'utiliser AMS.

Files d'attente IBM MQ traitées par le pont IMS
Les files d'attente IBM MQ traitées par le pont IMS ne sont pas prises en charge.

Remarque : AMS est pris en charge sur les files d'attente de pont CICS . Vous devez utiliser le même ID utilisateur pour MQPUT (chiffrement) et MQGET (déchiffrement) sur les files d'attente de pont CICS .

Insertion dans la méthode d'accès get en attente
L'opération put to waiting getter n'est pas prise en charge pour les applications getter sur les files d'attente pour lesquelles des règles AMS sont définies.

V 9.1.3 **Interception MCA de serveur à serveur**
Depuis IBM MQ 9.1.3, sous IBM MQ for z/OS, l'interception MCA de serveur à serveur n'est prise en charge que pour les types de canal émetteur, serveur, récepteur et demandeur.

- Les utilisateurs doivent éviter de placer plusieurs certificats avec le même nom distinctif dans un même fichier de clés, car le choix du certificat à utiliser lors de la protection d'un message n'est pas défini.
- AMS n'est pas pris en charge dans JMS si la propriété **WMQ_PROVIDER_VERSION** est définie sur 6.
- L'intercepteur AMS n'est pas pris en charge pour les canaux AMQP ou MQTT.

V 9.1.3 z/OS Présentation de l'interception Advanced Message Security sur les canaux de transmission de messages

Sous z/OS, l'interception Advanced Message Security (AMS) améliore l'offre existante en ajoutant une option supplémentaire de protection des règles de sécurité (SPLPROT) aux canaux émetteur, serveur, récepteur et demandeur.

Actuellement, à l'aide de l'exemple d'une chambre de compensation communiquant avec une banque, les deux côtés du système doivent prendre en charge AMS, comme illustré dans la Figure 1.

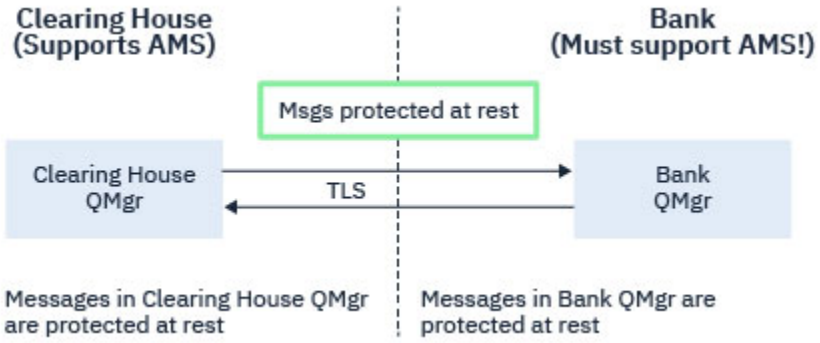


Figure 32. Utilisation actuelle d'AMS

Un avantage clé de l'option supplémentaire est que si votre entreprise a configuré AMS et que tous vos partenaires commerciaux ne prennent pas en charge AMS, vous pouvez supprimer la protection des messages sortants et protéger les messages entrants sur les canaux vers et depuis les partenaires commerciaux qui ne prennent pas en charge AMS.

A l'aide de l'exemple d'une chambre de compensation et de banques, ce scénario est illustré dans la figure 2, où il existe un flux de messages entre la chambre de compensation, les banques et les partenaires commerciaux dans lesquels certaines institutions ont AMSet d'autres non.

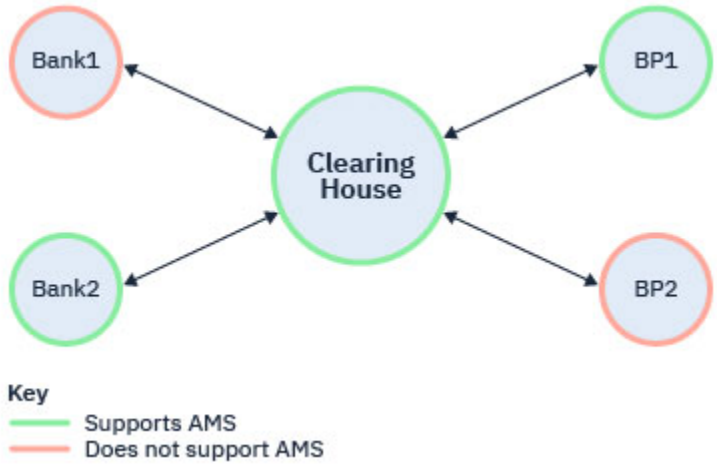


Figure 33. Certains partenaires soutiennent l'AMS et d'autres ne le font pas

Généralement, les canaux sont activés pour TLS.

Toutefois, il peut arriver que certaines banques et certains partenaires commerciaux ne prennent pas en charge AMS, et qu'il soit nécessaire de pouvoir échanger des messages entre toutes les banques et les partenaires commerciaux. Ce scénario est illustré dans la Figure 3

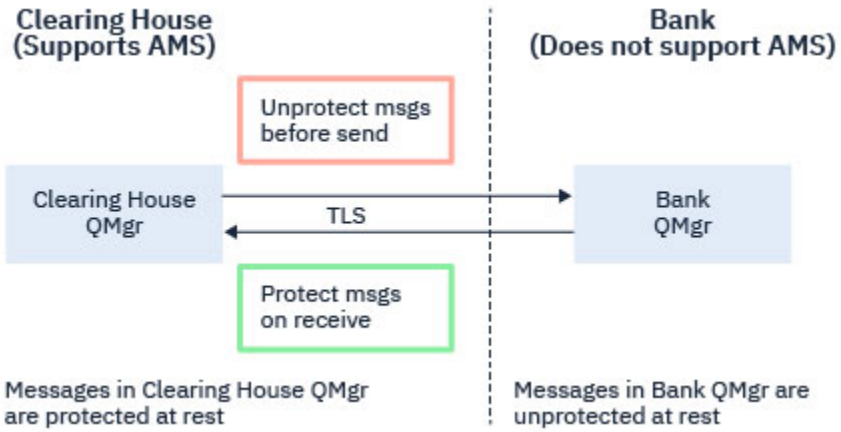


Figure 34. Flux de messages entre les partenaires commerciaux

Tâches associées

Exemples de configuration d'interception de canal de transmission de messages inter-serveurs

V 9.1.3 z/OS *Interception AMS sur les canaux de messages inter-serveurs*

L'interception de canal de message serveur à serveur permet de contrôler si des règles Advanced Message Security (AMS) applicables doivent être appliquées aux messages, lorsque les agents de canal de message de type émetteur extraient des messages des files d'attente de transmission et que les agents de canal de message de type récepteur placent des messages dans des files d'attente cible.

Cela permet d'activer la protection AMS sur un gestionnaire de files d'attente lors de la communication, à l'aide de canaux de message serveur à serveur de type émetteur, serveur, récepteur et demandeur, avec un gestionnaire de files d'attente pour lequel AMS n'est pas activé.

En d'autres termes, les messages protégés AMS dans les gestionnaires de files d'attente activés pour AMS peuvent être non protégés avant d'être envoyés à des gestionnaires de files d'attente non activés pourAMS , et les messages non protégés reçus des gestionnaires de files d'attente non activés pourAMS peuvent être protégés, par les règles AMS applicables, sur les gestionnaires de files d'attente activés pour AMS .

Configuration de l'interception des canaux de transmission de messages de serveur à serveur

L'interception de canal de message serveur à serveur est configurée avec l'attribut SPLPROT sur les canaux de type émetteur, serveur, récepteur ou demandeur. Les options disponibles pour configurer le comportement dépendent du type de canal spécifié:

PASSTHRU
Transmission sans modification de tout message envoyé ou reçu par l'agent MCA pour ce canal.

Cette valeur est valide pour les canaux avec un type de canal (**CHLTYPE**) de SDR, SVR, RCR ou RQSTR, et est la valeur par défaut.

REMOVE
Retrait de toute protection AMS dans les messages extraits de la file d'attente de transmission par l'agent MCA et envoi des messages au partenaire.

Lorsque l'agent MCA obtient un message de la file d'attente de transmission, si une stratégie AMS est définie pour la file d'attente de transmission, elle est appliquée afin de retirer toute protection AMS dans le message avant son envoi via le canal. Si aucune stratégie AMS n'est définie pour la file d'attente de transmission, le message est envoyé tel quel.

Cette valeur est valide uniquement pour les canaux dont le type est SDR ou SVR.

ASPOLICY

En fonction de la stratégie définie pour la file d'attente cible, application de la protection AMS aux messages entrants avant leur placement dans la file d'attente cible.

Lorsque l'agent MCA reçoit un message entrant, si une stratégie AMS est définie pour la file d'attente cible, la protection AMS est appliquée au message avant son placement dans la file d'attente cible. Si une règle AMS n'est pas définie pour la file d'attente cible, le message est placé dans la file d'attente cible comme c'est le cas.

Cette valeur est valide uniquement pour les canaux dont le type est RCVR ou RQSTR.

ID utilisateur pour l'interception de canal de transmission de messages

Les exigences relatives aux ID utilisateur utilisés avec l'interception des canaux de transmission de messages de serveur à serveur sont les mêmes que pour les applications AMS existantes. Pour un canal en cours d'exécution, l'agent de canal de transmission de messages extrait les messages d'une file d'attente de transmission et l'agent de canal de transmission de messages de réception insère les messages dans les files d'attente cible. La zone ID utilisateur de l'agent de canal de transmission de messages (MCAUSER), définie sur les canaux serveur à serveur, définit l'ID utilisateur sous lequel les agents de canal de transmission de messages exécutent les requêtes d'insertion et d'obtention.

Avec l'interception de canal de message serveur à serveur, les fonctions AMS sont exécutées lors des demandes d'extraction et d'insertion, comme pour les autres applications activées par AMS . Par conséquent, les ID utilisateur de l'agent MCA ont les mêmes exigences que celles applicables aux ID utilisateur d'application AMS .

L'utilisateur MCAUSER utilisé pour effectuer l'insertion et l'extraction est configurable et dépend du fait qu'il s'agisse d'un canal sortant ou entrant. Voir MCAUSER pour plus de détails sur la manière dont l'ID utilisateur choisi effectue des actions sur l'agent MCA. Par conséquent, l'ID utilisateur sous lequel l'initiateur de canal s'exécute est l'ID utilisateur qui doit être utilisé pour les fonctions AMS exécutées lors de l'interception de canal de message de serveur à serveur. Par conséquent, ces ID utilisateur ont les mêmes exigences que celles applicables aux ID utilisateur d'application AMS .

L'authentification est effectuée à l'aide des règles existantes pour le canal détaillé pour les canaux avec la configuration PUTAUT. Pour plus d'informations, voir ID utilisateur utilisés par l'initiateur de canal .

Remarque : L'interception de canal de message serveur à serveur ne prend pas en compte la valeur de l'attribut de canal PUTAUT.

Taille des messages et MAXMSGL

En raison de la protection AMS , la taille des messages protégés sera supérieure à la taille des messages d'origine.

Les messages protégés sont plus volumineux que les messages non protégés. Par conséquent, la valeur de l'attribut **MAXMSGL** , à la fois sur les files d'attente et sur les canaux, peut avoir besoin d'être modifiée pour tenir compte de la taille des messages protégés.

Référence associée

Exemples de configuration d'interception de canal de transmission de messages inter-serveurs

Traitement des erreurs

IBM MQ Advanced Message Security définit une file d'attente de traitement des erreurs pour gérer les messages contenant des erreurs ou des messages qui ne peuvent pas être protégés.

Les messages défectueux sont traités comme des cas exceptionnels. Si un message reçu ne répond pas aux exigences de sécurité de la file d'attente dans laquelle il se trouve, par exemple, si le message est signé alors qu'il doit être chiffré, ou si le déchiffrement ou la vérification de la signature échoue, le message est envoyé à la file d'attente de traitement des erreurs. Un message peut être envoyé à la file d'attente de traitement des erreurs pour les raisons suivantes:

- Non-concordance de la qualité de protection-Il existe une non-concordance de la qualité de protection (QOP) entre le message reçu et la définition QOP dans la règle de sécurité.
- Erreur de déchiffrement-le message ne peut pas être déchiffré.
- Erreur d'en-tête PDMQ-L'en-tête de message Advanced Message Security (AMS) est inaccessible.
- Non-concordance de taille-la longueur d'un message après déchiffrement est différente de celle attendue.
- Non-concordance de la force de l'algorithme de chiffrement-l'algorithme de chiffrement du message est plus faible que requis.
- Erreur inconnue-une erreur inattendue s'est produite.

AMS utilise SYSTEM.PROTECTION.ERROR.QUEUE comme file d'attente de traitement des erreurs. Tous les messages insérés par IBM MQ AMS dans SYSTEM.PROTECTION.ERROR.QUEUE sont précédées d'un en-tête MQDLH.

Votre administrateur IBM MQ peut également définir le système SYSTEM.PROTECTION.ERROR.QUEUE en tant que file d'attente alias pointant vers une autre file d'attente.

V 9.1.3 z/OS Depuis IBM MQ 9.1.3, sous IBM MQ for z/OS, si l'interception MCA (Message Channel Agent) de serveur à serveur est utilisée:

- Si, pour l'une des raisons précédemment indiquées, IBM MQ AMS déplace les messages de la file d'attente de transmission vers la file d'attente de traitement des erreurs, l'agent MCA émetteur traite simplement le prochain message disponible dans la file d'attente de transmission.
- En général, les règles de canal existantes s'appliquent pour:
  - Insertion de messages dans la file d'attente des messages non livrés, et
  - Les actions effectuées si les insertions dans la file d'attente des messages non livrés doivent échouer.

Pour plus d'informations sur des scénarios spécifiques, voir «Messages non distribués pour AMS sous z/OS», à la page 589 .

V 9.1.3 z/OS *Messages non distribués pour AMS sous z/OS*

Scénarios spécifiques liés à l'interception de l'agent de canal de message serveur à serveur sur IBM MQ for z/OS.

Depuis IBM MQ 9.1.3, sous IBM MQ for z/OS, si l'interception MCA (Message Channel Agent) de serveur à serveur est utilisée:

- Si, après avoir reçu et protégé un message, l'agent MCA émetteur ne parvient pas à distribuer un message pour une raison quelconque, par exemple parce que le message est trop volumineux pour le canal, si l'attribut de canal émetteur USEDLQ est défini sur YES, l'agent MCA émetteur déplace le message vers la file d'attente des messages non livrés (DLQ) locale.

  Si SYSTEM.DEAD.LETTER.QUEUE est utilisée en tant que file d'attente DLQ locale, le message est placé sans protection.

  Remarque : IBM MQ AMS ne prend pas en charge la protection des messages insérés dans les files d'attente du système.

  Si une file d'attente DLQ nommée est utilisée comme file d'attente DLQ locale, le message sera placé protégé si vous avez défini une stratégie IBM MQ AMS portant le même nom que la file d'attente DLQ nommée et non protégée si vous n'avez pas défini de stratégie appropriée.
- Si un message ne peut pas être inséré dans le DLQ local pour une raison quelconque, si NPMSPEED du canal est défini sur NORMAL ou que le message est un message persistant, le lot de messages en cours est annulé et le canal passe à l'état RETRY. Sinon, le message est supprimé et l'agent MCA émetteur continue à traiter le message suivant dans la file d'attente de transmission.
- Etant donné que les règles de sécurité n'ont aucun effet sur SYSTEM.DEAD.LETTER.QUEUE, ou les autres files d'attente SYSTEM répertoriées dans «Protection des files d'attente système dans AMS», à la page 663, si SYSTEM.DEAD.LETTER.QUEUE est en cours d'utilisation, les messages insérés dans cette

file d'attente par les MCM sont placés en l'état. Autrement dit, si des messages ont déjà été protégés, ils sont placés protégés ; dans le cas contraire, ils sont placés non protégés.

Si l'attribut DEADQ du gestionnaire de files d'attente a été défini sur le nom d'une autre file d'attente de rebut (non système) et qu'il n'existe pas de règle AMS portant le même nom, les messages placés dans cette file d'attente par les agents MCA sont placés en l'état. Autrement dit, si des messages ont déjà été protégés, ils sont placés protégés ; dans le cas contraire, ils sont placés non protégés.

Si l'attribut DEADQ du gestionnaire de files d'attente a été défini sur le nom d'une autre file d'attente de rebut (non système) et qu'il existe une règle AMS portant le même nom que la file d'attente des messages non livrés, cette règle est utilisée pour protéger les messages insérés dans cette file d'attente par les agents MCA. Si le message a déjà été protégé, il n'est pas protégé à nouveau ; cela permet d'éviter une double protection. S'il n'existe pas de règle AMS portant le même nom, les messages sont placés en tant que tels.

- S'il existe une règle pour le DLQ avec l'option de tolérance dans la commande setmqspl définie sur off, c'est-à-dire'**-t** O', l'insertion dans le DLQ échoue si le message n'est pas AMS protégé et qu'il n'a donc pas d'en-tête PDMQ. Cela se produit si le message arrive au destinataire sans en-tête PDMQ. C'est-à-dire que le putter d'origine du message n'avait pas de règle pour la destination et que le récepteur n'a pas de SPLPROT (ASPOLICY) défini.
- Il se peut qu'un agent MCA n'arrive pas à insérer un message dans la file d'attente des messages non livrés si la règle AMS définie pour la file d'attente des messages non livrés n'autorise pas l'ID utilisateur sous lequel l'initiateur de canal s'exécute pour protéger le message.
- Les canaux récepteurs placent généralement les messages non distribués dans le DLQ local, tandis que les canaux émetteurs placent généralement les messages qui ne peuvent pas être traités pour une raison quelconque, par exemple, un message trop volumineux pour la file d'attente ou un en-tête MQXQH incorrect, et ainsi de suite dans le DLQ local.
- Les gestionnaires DLQ ne regardent généralement que l'en-tête DLQ (DLH) et non la charge de message elle-même. Par conséquent, le fait que la charge de message puisse être protégée n'empêche pas les gestionnaires de déterminer la raison pour laquelle le message a été placé sur la file d'attente des messages non livrés.
- Si un DLQ n'est pas défini, le canal:
  - Se termine de manière anormale (et passe à l'état de relance) si un message persistant ne peut pas être distribué.
  - Supprime un message non persistant non distribué et continue à s'exécuter.

Concepts associés

«Traitement des erreurs», à la page 588
IBM MQ Advanced Message Security définit une file d'attente de traitement des erreurs pour gérer les messages contenant des erreurs ou des messages qui ne peuvent pas être protégés.

Scénarios utilisateur

Familiarisez-vous avec les scénarios possibles pour comprendre les objectifs métier que vous pouvez atteindre avec Advanced Message Security.

Windows *Guide de démarrage rapide pour AMS sur les plateformes Windows*

Utilisez ce guide pour configurer rapidement Advanced Message Security afin d'assurer la sécurité des messages sur les plateformes Windows . Lorsque vous l'aurez terminé, vous aurez créé une base de données de clés pour vérifier les identités utilisateur et défini des règles de signature / chiffrement pour votre gestionnaire de files d'attente.

Avant de commencer

Au moins les fonctions suivantes doivent être installées sur votre système:

- serveur

- Kit d'outils de développement (pour les exemples de programme)
- Advanced Message Security

Pour plus d'informations, voir FonctionsIBM MQ pour les systèmes Windows .

Pour plus d'informations sur l'utilisation de la commande **setmqenv** pour initialiser l'environnement en cours afin que les commandes IBM MQ appropriées puissent être localisées et exécutées par le système d'exploitation, voir setmqenv (set IBM MQ environment).

1. Création d'un gestionnaire de files d'attente et d'une file d'attente

Pourquoi et quand exécuter cette tâche

Tous les exemples suivants utilisent une file d'attente nommée TEST.Q pour la transmission de messages entre les applications. Advanced Message Security utilise des intercepteurs pour signer et chiffrer les messages lorsqu'ils entrent dans l'infrastructure IBM MQ via l'interface IBM MQ standard. La configuration de base est effectuée dans IBM MQ et est configurée dans les étapes suivantes.

Vous pouvez utiliser IBM MQ Explorer pour créer le gestionnaire de files d'attente QM_VERIFY_AMS et sa file d'attente locale appelée TEST.Q à l'aide de tous les paramètres de l'assistant par défaut, ou vous pouvez utiliser les commandes disponibles dans C:\Program Files\IBM\MQ\bin. N'oubliez pas que vous devez être membre du groupe d'utilisateurs mqm pour exécuter les commandes d'administration suivantes.

Procédure

1. Création d'un gestionnaire de files d'attente

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Démarrer le gestionnaire de files d'attente

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Créez une file d'attente appelée TEST.Q en entrant la commande suivante dans **runmqsc** pour le gestionnaire de files d'attente QM_VERIFY_AMS

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Résultats

Si la procédure est terminée, la commande entrée dans **runmqsc** affiche les détails relatifs à TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Création et autorisation d'utilisateurs

Pourquoi et quand exécuter cette tâche

Deux utilisateurs apparaissent dans cet exemple: alice, l'expéditeur et bob, le destinataire. Pour utiliser la file d'attente d'application, ces utilisateurs doivent être autorisés à l'utiliser. De même, pour utiliser correctement les règles de protection que nous définirons, ces utilisateurs doivent être autorisés à accéder à certaines files d'attente du système. Pour plus d'informations sur la commande **setmqaut** , voir **setmqaut**.

Procédure

1. Créez les deux utilisateurs et assurez-vous que HOMEPATH et HOMEDRIVE sont définis pour ces deux utilisateurs.

2. Autoriser les utilisateurs à se connecter au gestionnaire de files d'attente et à utiliser la file d'attente

```
setmqaut -m QM_VERIFY_AMS -t qmgr -p alice -p bob +connect +inq
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p alice +put
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p bob +get
```

3. Vous devez également autoriser les deux utilisateurs à parcourir la file d'attente de règles système et à placer des messages dans la file d'attente d'erreurs.

```
setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.POLICY.QUEUE -p alice -p bob +browse
setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.ERROR.QUEUE -p alice -p bob +put
```

Avertissement : IBM MQ optimise les performances en mettant en cache les règles de sorte que vous n'ayez pas à parcourir les enregistrements pour obtenir des détails sur les règles dans SYSTEM.PROTECTION.POLICY.QUEUE dans tous les cas.

IBM MQ ne met pas en cache toutes les règles disponibles. S'il existe un nombre élevé de règles, IBM MQ met en cache un nombre limité de règles. Par conséquent, si le nombre de règles définies pour le gestionnaire de files d'attente est faible, il n'est pas nécessaire de fournir l'option de navigation à SYSTEM.PROTECTION.POLICY.QUEUE.

Toutefois, vous devez accorder le droit de navigation à cette file d'attente, au cas où un nombre élevé de règles serait défini, ou si vous utilisez d'anciens clients. SYSTEM.PROTECTION.ERROR.QUEUE est utilisé pour placer les messages d'erreur générés par le code AMS. Le droit d'insertion sur cette file d'attente est vérifié uniquement lorsque vous tentez d'insérer un message d'erreur dans la file d'attente. Votre droit d'insertion sur la file d'attente n'est pas vérifié lorsque vous tentez d'insérer ou d'extraire un message d'une file d'attente protégée AMS.

Résultats

Les utilisateurs sont maintenant créés et les droits requis leur sont accordés.

Que faire ensuite

Pour vérifier si les étapes ont été effectuées correctement, utilisez les exemples `amqsput` et `amqsget` comme décrit dans la section «7. Test de la configuration», à la page 595.

3. Création d'une base de données de clés et de certificats

Pourquoi et quand exécuter cette tâche

L'intercepteur requiert la clé publique des utilisateurs qui l'envoient pour chiffrer le message. Par conséquent, la base de données de clés des identités utilisateur mappées aux clés publiques et privées doit être créée. Dans le système réel, où les utilisateurs et les applications sont dispersés sur plusieurs ordinateurs, chaque utilisateur dispose de son propre magasin de clés privé. De même, dans ce guide, nous créons des bases de données de clés pour `alice` et bob et nous partageons les certificats d'utilisateur entre eux.

Remarque : Dans ce guide, nous utilisons des exemples d'applications écrits en C se connectant à l'aide de liaisons locales. Si vous prévoyez d'utiliser des applications Java à l'aide de liaisons client, vous devez créer un magasin de clés JKS et des certificats à l'aide de la commande **`keytool`** , qui fait partie de l'environnement d'exécution Java (voir «Guide de démarrage rapide pour AMS avec les clients Java», à la page 614 pour plus de détails). Pour tous les autres langages et pour les applications Java utilisant des liaisons locales, les étapes de ce guide sont correctes.

Procédure

1. Utilisez l'interface graphique de gestion des clés IBM (`strmqikm.exe`) pour créer une nouvelle base de données de clés pour l'utilisateur `alice`.

```
Type:       CMS
Filename:   alicekey.kdb
Location:   C:/Documents and Settings/alice/AMS
```

Remarque :

- Il est conseillé d'utiliser un mot de passe fiable pour sécuriser la base de données.
- Vérifiez que la case **Stocker le mot de passe dans un fichier** est cochée.

2. Remplacez la vue de contenu de la base de données de clés par **Certificats personnels**.
3. Sélectionnez **New Self Signed** ; des certificats autosignés sont utilisés dans ce scénario.
4. Créez un certificat identifiant l'utilisateur `alice` à utiliser dans le chiffrement, à l'aide des zones suivantes:

```
Key label: Alice_Cert
Common Name: alice
Organisation: IBM
Country: GB
```

Remarque :

- Pour les besoins de ce guide, nous utilisons un certificat autosigné qui peut être créé sans utiliser d'autorité de certification. Pour les systèmes de production, il est conseillé de ne pas utiliser de certificats autosignés, mais de s'appuyer sur des certificats signés par une autorité de certification.
- Le paramètre **`Key label`** indique le nom du certificat, que les intercepteurs recherchent pour recevoir les informations nécessaires.
- Le paramètre **`Common Name`** et les paramètres facultatifs spécifient les détails du **nom distinctif** (DN), qui doit être unique pour chaque utilisateur.

5. Répétez les étapes 1 à 4 pour l'utilisateur bob

Résultats

Les deux utilisateurs `alice` et bob possèdent chacun un certificat autosigné.

4. Création de keystore.conf

Pourquoi et quand exécuter cette tâche

Vous devez pointer les intercepteurs Advanced Message Security vers le répertoire dans lequel se trouvent les bases de données de clés et les certificats located.This est effectuée via le fichier `keystore.conf` , qui contient ces informations sous forme de texte en clair. Chaque utilisateur doit disposer d'un fichier `keystore.conf` distinct dans le dossier `.mqs` . Cette étape doit être effectuée pour `alice` et bob.

Le contenu de `keystore.conf` doit être au format suivant:

```
cms.keystore = dir/keystore_file
cms.certificate = certificate_label
```

Exemple

Pour ce scénario, le contenu de `keystore.conf` sera le suivant:

```
cms.keystore = C:/Documents and Settings/alice/AMS/alicekey
cms.certificate = Alice_Cert
```

Remarque :

- Le chemin d'accès au fichier de clés doit être fourni sans inclure l'extension de fichier.

- Le label de certificat peut inclure des espaces, ainsi "Alice_Cert" et "Alice_Cert" (avec un espace à la fin) par exemple, sont reconnus comme des libellés de deux certificats différents. Cependant, pour éviter toute confusion, il est préférable de ne pas utiliser d'espaces dans le nom de l'étiquette.
- Il existe les formats de fichier de clés suivants: CMS (Cryptographic Message Syntax), JKS (Java Keystore) et JCEKS (Java Cryptographic Extension Keystore). Pour plus d'informations, voir «Structure du fichier de configuration du magasin de clés (keystore.conf) pour AMS», à la page 627.
- *%HOMEDRIVE%\%HOMEPATH%*\.mqs\keystore.conf (par exemple, C:\Documents and Settings\alice\.mqs\keystore.conf) est l'emplacement par défaut où Advanced Message Security recherche le fichier keystore.conf . Pour plus d'informations sur l'utilisation d'un emplacement autre que celui par défaut pour le keystore.conf, voir «Utilisation de magasins de clés et de certificats», à la page 626.
- Pour créer le répertoire .mqs , vous devez utiliser l'invite de commande.

5. Partage de certificats

Pourquoi et quand exécuter cette tâche

Partagez les certificats entre les deux bases de données de clés afin que chaque utilisateur puisse identifier l'autre. Cette opération est effectuée en extrayant le certificat public de chaque utilisateur dans un fichier, qui est ensuite ajouté à la base de données de clés de l'autre utilisateur.

Remarque : Prenez soin d'utiliser l'option *extract* et non l'option *export* . *Extract* obtient la clé publique de l'utilisateur, tandis que *export* obtient à la fois la clé publique et la clé privée. L'utilisation de *export* par erreur compromettrait complètement votre application en transmettant sa clé privée.

Procédure

1. Extrayez le certificat identifiant alice dans un fichier externe:

```
runmqakm -cert -extract -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
-label Alice_Cert -target alice_public.arm
```

2. Ajoutez le certificat au magasin de clés bob's :

```
runmqakm -cert -add -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd -label
Alice_Cert -file alice_public.arm
```

3. Répétez les étapes pour bob:

```
runmqakm -cert -extract -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd
-label Bob_Cert -target bob_public.arm

runmqakm -cert -add -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
-label Bob_Cert -file bob_public.arm
```

Résultats

Les deux utilisateurs alice et bob sont désormais en mesure de s'identifier mutuellement en ayant créé et partagé des certificats autosignés.

Que faire ensuite

Vérifiez qu'un certificat se trouve dans le magasin de clés en le parcourant à l'aide de l'interface graphique ou en exécutant les commandes suivantes qui impriment ses détails:

```
runmqakm -cert -details -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd -label
Alice_Cert
```

```
runmqakm -cert -details -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
-label Bob_Cert
```

6. Définition de la règle de file d'attente

Pourquoi et quand exécuter cette tâche

Avec le gestionnaire de files d'attente créé et les intercepteurs préparés pour intercepter les messages et les clés de chiffrement d'accès, nous pouvons commencer à définir des règles de protection sur `QM_VERIFY_AMS` à l'aide de la commande `setmqspl` . Pour plus d'informations sur cette commande, voir setmqspl . Chaque nom de règle doit être identique au nom de la file d'attente à laquelle il doit être appliqué.

Exemple

Voici un exemple de règle définie pour la file d'attente `TEST.Q` . Dans l'exemple, les messages sont signés avec l'algorithme SHA1 et chiffrés avec l'algorithme AES256 . `alice` est le seul émetteur valide et bob est le seul récepteur des messages de cette file d'attente:

```
setmqspl -m QM_VERIFY_AMS -p TEST.Q -s SHA1 -a "CN=alice,O=IBM,C=GB" -e AES256 -r
"CN=bob,O=IBM,C=GB"
```

Remarque : Les noms distinctifs correspondent exactement à ceux spécifiés dans le certificat de l'utilisateur respectif de la base de données de clés.

Que faire ensuite

Pour vérifier la règle que vous avez définie, exécutez la commande suivante:

```
dspmqspl -m QM_VERIFY_AMS
```

Pour imprimer les détails de la règle sous la forme d'un ensemble de commandes `setmqspl` , utilisez l'indicateur `-export` . Cela permet de stocker des règles déjà définies:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Test de la configuration

Pourquoi et quand exécuter cette tâche

En exécutant différents programmes sous différents utilisateurs, vous pouvez vérifier si l'application a été correctement configurée.

Procédure

1. Changez d'utilisateur pour qu'il s'exécute en tant qu'utilisateur `alice`

   Cliquez avec le bouton droit de la souris sur `cmd.exe` et sélectionnez **Exécuter en tant que ...**. Lorsque vous y êtes invité, connectez-vous en tant que `alice`.
2. En tant qu'utilisateur `alice` , placez un message à l'aide d'un exemple d'application:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```
3. Entrez le texte du message, puis appuyez sur Entrée.
4. Changez d'utilisateur pour qu'il s'exécute en tant qu'utilisateur bob

   Ouvrez une autre fenêtre en cliquant avec le bouton droit de la souris sur `cmd.exe` et en sélectionnant **Exécuter en tant que ...**. Lorsque vous y êtes invité, connectez-vous en tant que bob.

5. En tant qu'utilisateur bob , obtenez un message à l'aide d'un exemple d'application:

```
amqsget TEST.Q QM_VERIFY_AMS
```

Résultats

Si l'application a été correctement configurée pour les deux utilisateurs, le message de l'utilisateur `alice` s'affiche lorsque bob exécute l'application d'obtention.

8. Test du chiffrement

Pourquoi et quand exécuter cette tâche

Pour vérifier que le chiffrement est effectué comme prévu, créez une file d'attente alias qui fait référence à la file d'attente d'origine TEST.Q. Cette file d'attente alias n'ayant pas de règle de sécurité, aucun utilisateur ne dispose des informations permettant de déchiffrer le message. Par conséquent, les données chiffrées sont affichées.

Procédure

1. A l'aide de la commande **runmqsc** sur le gestionnaire de files d'attente QM_VERIFY_AMS, créez une file d'attente alias.

```
DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
```

2. Accordez à bob l'accès pour parcourir la file d'attente alias

```
setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
```

3. En tant qu'utilisateur `alice`, placez un autre message à l'aide d'un exemple d'application comme précédemment:

```
amqsput TEST.Q QM_VERIFY_AMS
```

4. En tant qu'utilisateur bob, parcourez le message à l'aide d'un exemple d'application via la file d'attente alias cette fois:

```
amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. En tant qu'utilisateur bob, obtenez le message à l'aide d'un exemple d'application à partir de la file d'attente locale:

```
amqsget TEST.Q QM_VERIFY_AMS
```

Résultats

La sortie de l'application `amqsbcg` affiche les données chiffrées qui se trouvent dans la file d'attente prouvant que le message a été chiffré.

UNIX *Guide de démarrage rapide pour AMS sur UNIX*

Utilisez ce guide pour configurer rapidement Advanced Message Security afin de fournir la sécurité des messages sur UNIX. Lorsque vous l'aurez terminé, vous aurez créé une base de données de clés pour vérifier les identités utilisateur et défini des règles de signature / chiffrement pour votre gestionnaire de files d'attente.

Avant de commencer

Au moins les composants suivants doivent être installés sur votre système:

- MQ Light
- serveur
- Exemples de programme
- IBM Global Security Kit
- Advanced Message Security

Reportez-vous aux rubriques suivantes pour connaître les noms de composant sur chaque plateforme spécifique:

- 

ComposantsIBM MQ pour les systèmes Linux
- 

ComposantsIBM MQ pour les systèmes AIX
- 

ComposantsIBM MQ pour les systèmes Solaris

1. Création d'un gestionnaire de files d'attente et d'une file d'attente

Pourquoi et quand exécuter cette tâche

Tous les exemples suivants utilisent une file d'attente nommée TEST.Q pour la transmission de messages entre les applications. Advanced Message Security utilise des intercepteurs pour signer et chiffrer les messages lorsqu'ils entrent dans l'infrastructure IBM MQ via l'interface IBM MQ standard. La configuration de base est effectuée dans IBM MQ et est configurée dans les étapes suivantes.

Vous pouvez utiliser IBM MQ Explorer pour créer le gestionnaire de files d'attente QM_VERIFY_AMS et sa file d'attente locale appelée TEST.Q à l'aide de tous les paramètres de l'assistant par défaut, ou vous pouvez utiliser les commandes disponibles dans *MQ_INSTALLATION_PATH*/bin. N'oubliez pas que vous devez être membre du groupe d'utilisateurs mqm pour exécuter les commandes d'administration suivantes.

Procédure

1. Création d'un gestionnaire de files d'attente

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Démarrer le gestionnaire de files d'attente

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Créez une file d'attente appelée TEST.Q en entrant la commande suivante dans **runmqsc** pour le gestionnaire de files d'attente QM_VERIFY_AMS

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Résultats

Si la procédure a abouti, la commande suivante entrée dans **runmqsc** affiche les détails relatifs à TEST.Q:

```
DISPLAY Q(TEST.Q)
```

Pourquoi et quand exécuter cette tâche

Deux utilisateurs apparaissent dans cet exemple: `alice`, l'expéditeur et bob, le destinataire. Pour utiliser la file d'attente d'application, ces utilisateurs doivent être autorisés à l'utiliser. De même, pour utiliser correctement les règles de protection que nous définirons, ces utilisateurs doivent être autorisés à accéder à certaines files d'attente du système. Pour plus d'informations sur la commande **setmqaut** , voir **setmqaut**.

Procédure

1. Créer les deux utilisateurs

```
useradd alice
useradd bob
```

2. Autoriser les utilisateurs à se connecter au gestionnaire de files d'attente et à utiliser la file d'attente

```
setmqaut -m QM_VERIFY_AMS -t qmgr -p alice -p bob +connect +inq
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p alice +put
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p bob +get
```

3. Vous devez également autoriser les deux utilisateurs à parcourir la file d'attente de règles système et à placer des messages dans la file d'attente d'erreurs.

```
setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.POLICY.QUEUE -p alice -p bob +browse
setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.ERROR.QUEUE -p alice -p bob +put
```

Avertissement : IBM MQ optimise les performances en mettant en cache les règles de sorte que vous n'ayez pas à parcourir les enregistrements pour obtenir des détails sur les règles dans SYSTEM.PROTECTION.POLICY.QUEUE dans tous les cas.

IBM MQ ne met pas en cache toutes les règles disponibles. S'il existe un nombre élevé de règles, IBM MQ met en cache un nombre limité de règles. Par conséquent, si le nombre de règles définies pour le gestionnaire de files d'attente est faible, il n'est pas nécessaire de fournir l'option de navigation à SYSTEM.PROTECTION.POLICY.QUEUE.

Toutefois, vous devez accorder le droit de navigation à cette file d'attente, au cas où un nombre élevé de règles serait défini, ou si vous utilisez d'anciens clients. SYSTEM.PROTECTION.ERROR.QUEUE est utilisé pour placer les messages d'erreur générés par le code AMS. Le droit d'insertion sur cette file d'attente est vérifié uniquement lorsque vous tentez d'insérer un message d'erreur dans la file d'attente. Votre droit d'insertion sur la file d'attente n'est pas vérifié lorsque vous tentez d'insérer ou d'extraire un message d'une file d'attente protégée AMS.

Résultats

Des groupes d'utilisateurs sont maintenant créés et les droits requis leur sont accordés. Ainsi, les utilisateurs affectés à ces groupes auront également le droit de se connecter au gestionnaire de files d'attente et d'insérer et d'extraire de la file d'attente.

Que faire ensuite

Pour vérifier si les étapes ont été effectuées correctement, utilisez les exemples `amqsput` et `amqsget` comme décrit dans la section «8. Test du chiffrement», à la page 602.

3. Création d'une base de données de clés et de certificats

Pourquoi et quand exécuter cette tâche

Pour chiffrer le message, l'intercepteur requiert la clé privée de l'utilisateur émetteur et la ou les clés publiques du ou des destinataires. Par conséquent, la base de données de clés des identités utilisateur mappées aux clés publiques et privées doit être créée. Dans le système réel, où les utilisateurs et les applications sont dispersés sur plusieurs ordinateurs, chaque utilisateur dispose de son propre magasin de clés privé. De même, dans ce guide, nous créons des bases de données de clés pour `alice` et bob et nous partageons les certificats d'utilisateur entre eux.

Remarque : Dans ce guide, nous utilisons des exemples d'applications écrits en C se connectant à l'aide de liaisons locales. Si vous prévoyez d'utiliser des applications Java à l'aide de liaisons client, vous devez créer un magasin de clés JKS et des certificats à l'aide de la commande **`keytool`** , qui fait partie de l'environnement d'exécution Java (voir «Guide de démarrage rapide pour AMS avec les clients Java», à la page 614 pour plus de détails). Pour tous les autres langages et pour les applications Java utilisant des liaisons locales, les étapes de ce guide sont correctes.

Procédure

1. Créer une nouvelle base de données de clés pour l'utilisateur `alice`

   ```
   mkdir /home/alice/.mqs -p
   runmqakm -keydb -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -stash
   ```

   Remarque :

   - Il est conseillé d'utiliser un mot de passe fiable pour sécuriser la base de données.
   - Le paramètre **`stash`** stocke le mot de passe dans le fichier `key.sth` , que les intercepteurs peuvent utiliser pour ouvrir la base de données.
2. Vérifiez que la base de données de clés est lisible

   ```
   chmod +r /home/alice/.mqs/alicekey.kdb
   ```
3. Créez un certificat identifiant l'utilisateur `alice` à utiliser dans le chiffrement

   ```
   runmqakm -cert -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd
   -label Alice_Cert -dn "cn=alice,O=IBM,c=GB" -default_cert yes
   ```

   Remarque :

   - Pour les besoins de ce guide, nous utilisons un certificat autosigné qui peut être créé sans utiliser d'autorité de certification. Pour les systèmes de production, il est conseillé de ne pas utiliser de certificats autosignés, mais de s'appuyer sur des certificats signés par une autorité de certification.
   - Le paramètre **`label`** indique le nom du certificat, que les intercepteurs recherchent pour recevoir les informations nécessaires.
   - Le paramètre **DN** spécifie les détails du **nom distinctif** (DN), qui doit être unique pour chaque utilisateur.
4. Maintenant que nous avons créé la base de données de clés, nous devons en définir la propriété et nous assurer qu'elle est illisible par tous les autres utilisateurs.

   ```
   chown alice /home/alice/.mqs/alicekey.kdb /home/alice/.mqs/alicekey.sth
   chmod 600 /home/alice/.mqs/alicekey.kdb /home/alice/.mqs/alicekey.sth
   ```
5. Répétez les étapes 1 à 4 pour l'utilisateur bob

Résultats

Les deux utilisateurs `alice` et bob possèdent chacun un certificat autosigné.

4. Création de keystore.conf

Pourquoi et quand exécuter cette tâche

Vous devez pointer les intercepteurs Advanced Message Security vers le répertoire dans lequel se trouvent les bases de données de clés et les certificats. Cette opération est effectuée via le fichier `keystore.conf` , qui contient ces informations sous forme de texte en clair. Chaque utilisateur doit disposer d'un fichier `keystore.conf` distinct dans le dossier `.mqs` . Cette étape doit être effectuée pour `alice` et bob.

Le contenu de `keystore.conf` doit être au format suivant:

```
cms.keystore = dir/keystore_file
cms.certificate = certificate_label
```

Exemple

Pour ce scénario, le contenu de `keystore.conf` sera le suivant:

```
cms.keystore = /home/alice/.mqs/alicekey
cms.certificate = Alice_Cert
```

Remarque :

- Le chemin d'accès au fichier de clés doit être fourni sans inclure l'extension de fichier.
- Il existe les formats de fichier de clés suivants: CMS (Cryptographic Message Syntax), JKS (Java Keystore) et JCEKS (Java Cryptographic Extension Keystore). Pour plus d'informations, voir «Structure du fichier de configuration du magasin de clés (keystore.conf) pour AMS», à la page 627.
- `HOME/.mqs/keystore.conf` est l'emplacement par défaut où Advanced Message Security recherche le fichier `keystore.conf` . Pour plus d'informations sur l'utilisation d'un emplacement autre que celui par défaut pour le `keystore.conf`, voir «Utilisation de magasins de clés et de certificats», à la page 626.

5. Partage de certificats

Pourquoi et quand exécuter cette tâche

Partagez les certificats entre les deux bases de données de clés afin que chaque utilisateur puisse identifier l'autre. Cette opération est effectuée en extrayant le certificat public de chaque utilisateur dans un fichier, qui est ensuite ajouté à la base de données de clés de l'autre utilisateur.

Remarque : Prenez soin d'utiliser l'option *extract* et non l'option *export* . *Extract* obtient la clé publique de l'utilisateur, tandis que *export* obtient à la fois la clé publique et la clé privée. L'utilisation de *export* par erreur compromettrait complètement votre application en transmettant sa clé privée.

Procédure

1. Extrayez le certificat identifiant `alice` dans un fichier externe:

   ```
   runmqakm -cert -extract -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Alice_Cert
   -target alice_public.arm
   ```

2. Ajoutez le certificat au magasin de clés `bob's` :

   ```
   runmqakm -cert -add -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Alice_Cert -file
   alice_public.arm
   ```

3. Répétez l'étape pour bob:

```
runmqakm -cert -extract -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Bob_Cert -target
bob_public.arm
```

4. Ajoutez le certificat pour bob au magasin de clés `alice's` :

```
runmqakm -cert -add -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Bob_Cert -file
bob_public.arm
```

Résultats

Les deux utilisateurs `alice` et bob sont désormais en mesure de s'identifier mutuellement en ayant créé et partagé des certificats autosignés.

Que faire ensuite

Vérifiez qu'un certificat se trouve dans le magasin de clés en exécutant les commandes suivantes qui impriment ses détails:

```
runmqakm -cert -details -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Alice_Cert
runmqakm -cert -details -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Bob_Cert
```

6. Définition de la règle de file d'attente

Pourquoi et quand exécuter cette tâche

Avec le gestionnaire de files d'attente créé et les intercepteurs préparés pour intercepter les messages et les clés de chiffrement d'accès, nous pouvons commencer à définir des règles de protection sur `QM_VERIFY_AMS` à l'aide de la commande `setmqspl` . Pour plus d'informations sur cette commande, voir setmqspl . Chaque nom de règle doit être identique au nom de la file d'attente à laquelle il doit être appliqué.

Exemple

Voici un exemple de règle définie pour la file d'attente `TEST.Q` . Dans cet exemple, les messages sont signés par l'utilisateur `alice` à l'aide de l'algorithme SHA1 et chiffrés à l'aide de l'algorithme AES 256 bits. `alice` est le seul émetteur valide et bob est le seul récepteur des messages de cette file d'attente:

```
setmqspl -m QM_VERIFY_AMS -p TEST.Q -s SHA1 -a "CN=alice,O=IBM,C=GB" -e AES256 -r
"CN=bob,O=IBM,C=GB"
```

Remarque : Les noms distinctifs correspondent exactement à ceux spécifiés dans le certificat de l'utilisateur respectif de la base de données de clés.

Que faire ensuite

Pour vérifier la règle que vous avez définie, exécutez la commande suivante:

```
dspmqspl -m QM_VERIFY_AMS
```

Pour imprimer les détails de la règle sous la forme d'un ensemble de commandes `setmqspl` , utilisez l'indicateur `-export` . Cela permet de stocker des règles déjà définies:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Test de la configuration

Pourquoi et quand exécuter cette tâche

En exécutant différents programmes sous différents utilisateurs, vous pouvez vérifier si l'application a été correctement configurée.

Procédure

1. Accédez au répertoire contenant les exemples. Si MQ est installé dans un emplacement autre que celui par défaut, il se peut qu'il se trouve à un autre emplacement.

```
cd /opt/mqm/samp/bin
```

2. Changez d'utilisateur pour qu'il s'exécute en tant qu'utilisateur `alice`

```
su alice
```

3. En tant qu'utilisateur `alice`, placez un message à l'aide d'un exemple d'application:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Entrez le texte du message, puis appuyez sur Entrée.
5. Arrêt de l'exécution en tant qu'utilisateur `alice`

```
exit
```

6. Changez d'utilisateur pour qu'il s'exécute en tant qu'utilisateur bob

```
su bob
```

7. En tant qu'utilisateur bob, obtenez un message à l'aide d'un exemple d'application:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Résultats

Si l'application a été correctement configurée pour les deux utilisateurs, le message de l'utilisateur `alice` s'affiche lorsque bob exécute l'application d'obtention.

8. Test du chiffrement

Pourquoi et quand exécuter cette tâche

Pour vérifier que le chiffrement est effectué comme prévu, créez une file d'attente alias qui fait référence à la file d'attente d'origine TEST.Q. Cette file d'attente alias n'ayant pas de règle de sécurité, aucun utilisateur ne dispose des informations permettant de déchiffrer le message. Par conséquent, les données chiffrées sont affichées.

Procédure

1. A l'aide de la commande **runmqsc** sur le gestionnaire de files d'attente QM_VERIFY_AMS, créez une file d'attente alias.

```
DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
```

2. Accordez à bob l'accès pour parcourir la file d'attente alias

```
setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
```

3. En tant qu'utilisateur `alice`, placez un autre message à l'aide d'un exemple d'application comme précédemment:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. En tant qu'utilisateur bob, parcourez le message à l'aide d'un exemple d'application via la file d'attente alias cette fois:

```
./amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. En tant qu'utilisateur bob, obtenez le message à l'aide d'un exemple d'application à partir de la file d'attente locale:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Résultats

La sortie de l'application `amqsbcg` affichera les données chiffrées qui se trouvent dans la file d'attente prouvant que le message a été chiffré.

z/OS *Exemples de configuration sous z/OS*

Cette section fournit des exemples de configuration de politiques et de certificats pour les scénarios de mise en file d'attente Advanced Message Security sur z/OS.

Voir Configuration de Advanced Message Security for z/OS pour plus de détails sur la configuration de Advanced Message Security.

Les exemples couvrent les règles Advanced Message Security requises et les certificats numériques qui doivent exister par rapport aux utilisateurs et aux fichiers de clés. Les exemples supposent que les utilisateurs impliqués dans les scénarios ont été configurés en suivant les instructions fournies dans Accorder aux utilisateurs des droits d'accès aux ressources pour Advanced Message Security.

V 9.1.3 De plus, à partir de IBM MQ 9.1.3 , voir Exemples d'interception de canal de message de serveur à serveur.

z/OS *Mise en file d'attente locale des messages protégés contre l'intégrité sur z/OS*

Cet exemple détaille les politiques et les certificats Advanced Message Security nécessaires pour envoyer et extraire des messages protégés contre l'intégrité vers et depuis une file d'attente, locale aux applications d'insertion et d'extraction.

Les exemples de gestionnaire de files d'attente et de file d'attente sont les suivants:

```
BNK6        - Queue manager
FIN.XFER.Q7 - Local queue
```

Ces utilisateurs sont utilisés:

```
WMQBNK6  - AMS task user
TELLER5  - Sending user
FINADM2  - Recipient user
```

Créer les certificats d'utilisateur

Dans cet exemple, un seul certificat d'utilisateur est requis. Il s'agit du certificat de l'utilisateur émetteur qui est nécessaire pour signer les messages protégés contre l'intégrité. L'utilisateur émetteur est'TELLER5'.

Le certificat de l'autorité de certification est également requis. Le certificat de l'autorité de certification est le certificat de l'autorité qui a émis le certificat de l'utilisateur. Il peut s'agir d'une chaîne de certificats. Si tel est le cas, tous les certificats de la chaîne sont requis dans le fichier de clés de l'utilisateur de la tâche Advanced Message Security , dans ce cas l'utilisateur WMQBNK6.

Un certificat d'autorité de certification peut être créé à l'aide de la commande RACDCERT RACF . Ce certificat est utilisé pour émettre des certificats d'utilisateur. Exemple :

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Cette commande RACDCERT crée un certificat d'autorité de certification qui peut ensuite être utilisé pour émettre un certificat d'utilisateur pour l'utilisateur'TELLER5'. Exemple :

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

Votre installation comporte des procédures pour choisir ou créer un certificat d'autorité de certification, ainsi que des procédures pour émettre des certificats et les distribuer aux systèmes appropriés.

Lors de l'exportation et de l'importation de ces certificats, Advanced Message Security requiert:

- Certificat de l'autorité de certification (chaîne).
- Certificat d'utilisateur et sa clé privée.

Si vous utilisez RACF, la commande RACDCERT EXPORT peut être utilisée pour exporter des certificats vers un fichier et la commande RACDCERT ADD peut être utilisée pour importer des certificats à partir du fichier. Pour plus d'informations sur ces commandes et d'autres commandes RACDCERT, voir *z/OS: Security Server RACF Command Language Reference*.

Dans ce cas, les certificats sont requis sur le système z/OS exécutant le gestionnaire de files d'attente BNK6.

Lorsque les certificats ont été importés sur le système z/OS exécutant BNK6, le certificat utilisateur requiert l'attribut TRUST. La commande RACDCERT ALTER peut être utilisée pour ajouter l'attribut TRUST au certificat. Exemple :

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
```

Dans cet exemple, aucun certificat n'est requis pour l'utilisateur destinataire.

Connexion de certificats à des fichiers de clés pertinents

Lorsque les certificats requis ont été créés ou importés et définis comme étant de confiance, ils doivent être connectés aux fichiers de clés utilisateur appropriés sur le système z/OS exécutant BNK6. Pour créer les fichiers de clés, utilisez les commandes RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur de la tâche Advanced Message Security , WMQBNK6, et un fichier de clés pour l'utilisateur émetteur,'TELLER5'. Notez que le nom de fichier de clés drq.ams.keyring est obligatoire et qu'il est sensible à la casse.

Une fois les fichiers de clés créés, les certificats appropriés peuvent être connectés:

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Le certificat d'utilisateur émetteur doit être connecté en tant que DEFAULT. Si l'utilisateur émetteur possède plusieurs certificats dans son fichier drq.ams.keyring, le certificat par défaut est utilisé à des fins de signature.

La création et la modification de certificats ne sont pas reconnues par Advanced Message Security tant que le gestionnaire de files d'attente n'est pas arrêté et redémarré ou que la commande z/OS **MODIFY** n'est pas utilisée pour actualiser la configuration de certificat Advanced Message Security . Exemple :

```
F BNK6AMSM,REFRESH KEYRING
```

Création de la règle Advanced Message Security

Dans cet exemple, les messages protégés contre l'intégrité sont placés dans la file d'attente FIN.XFER.Q7 par une application s'exécutant en tant qu'utilisateur'TELLER5'et extraite de la même file d'attente par une application s'exécutant en tant qu'utilisateur'FINADM2', de sorte qu'une seule règle Advanced Message Security est requise.

Les règles Advanced Message Security sont créées à l'aide de l'utilitaire CSQ0UTIL documenté à l'adresse The message security policy utility (CSQ0UTIL).

Utilisez l'utilitaire CSQ0UTIL pour exécuter la commande suivante:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK6. Le nom de la règle et la file d'attente associée sont FIN.XFER.Q7. L'algorithme utilisé pour générer la signature de l'expéditeur est MD5et le nom distinctif (DN) de l'utilisateur émetteur est'CN=Teller5,O=BCO,C=US'.

Après avoir défini la règle, redémarrez le gestionnaire de files d'attente BNK6 ou utilisez la commande z/OS **MODIFY** pour actualiser la configuration de la règle Advanced Message Security . Exemple :

```
F BNK6AMSM,REFRESH POLICY
```

z/OS *Mise en file d'attente locale des messages protégés par la confidentialité sur z/OS*

Cet exemple détaille les politiques et les certificats Advanced Message Security nécessaires pour envoyer et extraire des messages protégés par la confidentialité vers et depuis une file d'attente, locale aux applications d'insertion et d'extraction. Les messages protégés par la confidentialité sont à la fois signés et chiffrés.

Les exemples de gestionnaire de files d'attente et de file d'attente locale sont les suivants:

```
BNK6        - Queue manager
FIN.XFER.Q8 - Local queue
```

Ces utilisateurs sont utilisés:

```
WMQBNK6  - AMS task user
TELLER5  - Sending user
FINADM2  - Recipient user
```

Les étapes de configuration de ce scénario sont les suivantes:

Créer les certificats d'utilisateur

Dans cet exemple, deux certificats d'utilisateur sont requis. Il s'agit du certificat de l'utilisateur émetteur qui est nécessaire pour signer les messages et du certificat de l'utilisateur destinataire qui est nécessaire pour chiffrer et déchiffrer les données du message. L'utilisateur émetteur est'TELLER5'et l'utilisateur destinataire est'FINADM2'.

Le certificat de l'autorité de certification est également requis.Le certificat de l'autorité de certification est le certificat de l'autorité qui a émis le certificat de l'utilisateur. Il peut s'agir d'une chaîne de certificats. Si tel est le cas, tous les certificats de la chaîne sont requis dans le fichier de clés de l'utilisateur de la tâche Advanced Message Security , dans ce cas l'utilisateur WMQBNK6.

Un certificat d'autorité de certification peut être créé à l'aide de la commande RACDCERT RACF . Ce certificat est utilisé pour émettre des certificats d'utilisateur. Exemple :

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Cette commande RACDCERT crée un certificat d'autorité de certification qui peut ensuite être utilisé pour émettre des certificats d'utilisateur pour les utilisateurs'TELLER5'et'FINADM2'. Exemple :

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

Votre installation comporte des procédures pour choisir ou créer un certificat d'autorité de certification, ainsi que des procédures pour émettre des certificats et les distribuer aux systèmes appropriés.

Lors de l'exportation et de l'importation de ces certificats, Advanced Message Security requiert:

- Certificat de l'autorité de certification (chaîne).
- Certificat d'utilisateur émetteur et sa clé privée.
- Certificat d'utilisateur du destinataire et sa clé privée.

Si vous utilisez RACF, la commande RACDCERT EXPORT peut être utilisée pour exporter des certificats vers un fichier et la commande RACDCERT ADD peut être utilisée pour importer des certificats à partir du fichier. Pour plus d'informations sur ces commandes et d'autres commandes RACDCERT, voir RACDCERT (Manage RACF digital certificates) dans le manuel *z/OS: Security Server RACF Command Language Reference*.

Dans ce cas, les certificats sont requis sur le système z/OS exécutant le gestionnaire de files d'attente BNK6.

Lorsque les certificats ont été importés sur le système z/OS exécutant BNK6, les certificats d'utilisateur requièrent l'attribut TRUST. La commande RACDCERT ALTER peut être utilisée pour ajouter l'attribut TRUST au certificat. Exemple :

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Connexion de certificats à des fichiers de clés pertinents

Lorsque les certificats requis ont été créés ou importés et définis comme étant de confiance, ils doivent être connectés aux fichiers de clés utilisateur appropriés sur le système z/OS exécutant BNK6. Pour créer les fichiers de clés, utilisez la commande RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur de la tâche Advanced Message Security et des fichiers de clés pour les utilisateurs d'envoi et de destinataire. Notez que le nom de fichier de clés drq.ams.keyring est obligatoire et qu'il est sensible à la casse.

Une fois les fichiers de clés créés, les certificats appropriés peuvent être connectés.

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Les certificats d'utilisateur d'envoi et de destinataire doivent être connectés sous la forme DEFAULT. Si l'un des utilisateurs possède plusieurs certificats dans son fichier drq.ams.keyring, le certificat par défaut est utilisé à des fins de signature et de déchiffrement.

Le certificat de l'utilisateur destinataire doit également être connecté au fichier de clés de l'utilisateur de la tâche Advanced Message Security avec USAGE (SITE). En effet, la tâche Advanced Message Security a besoin de la clé publique du destinataire lors du chiffrement des données de message. USAGE (SITE) empêche la clé privée d'être accessible dans le fichier de clés.

La création et la modification de certificats ne sont pas reconnues par Advanced Message Security tant que le gestionnaire de files d'attente n'est pas arrêté et redémarré ou que la commande z/OS **MODIFY** n'est pas utilisée pour actualiser la configuration de certificat Advanced Message Security . Exemple :

```
F BNK6AMSM,REFRESH KEYRING
```

Création de la règle Advanced Message Security

Dans cet exemple, les messages protégés par la confidentialité sont placés dans la file d'attente FIN.XFER.Q8 par une application s'exécutant en tant qu'utilisateur'TELLER5'et extraite de la même file d'attente par une application s'exécutant en tant qu'utilisateur'FINADM2', de sorte qu'une seule règle Advanced Message Security est requise.

Les règles Advanced Message Security sont créées à l'aide de l'utilitaire CSQ0UTIL documenté à l'adresse The message security policy utility (CSQ0UTIL).

Utilisez l'utilitaire CSQ0UTIL pour exécuter la commande suivante:

```
setmqspl -m BNK6 -p FIN.XFER.Q8 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK6. Le nom de la règle et la file d'attente associée sont FIN.XFER.Q8. L'algorithme utilisé pour générer la signature de l'expéditeur est SHA1et le nom distinctif (DN) de l'utilisateur émetteur est'CN=Teller5,O=BCO,C=US', et l'utilisateur

destinataire est'CN=FinAdm2,O=BCO,C=US'. L'algorithme utilisé pour chiffrer les données de message est 3DES.

Après avoir défini la règle, redémarrez le gestionnaire de files d'attente BNK6 ou utilisez la commande z/OS **MODIFY** pour actualiser la configuration de la règle Advanced Message Security . Exemple :

```
F BNK6AMSM,REFRESH POLICY
```

z/OS *Mise en file d'attente distante des messages protégés contre l'intégrité sur z/OS*

Cet exemple détaille les règles et les certificats Advanced Message Security nécessaires pour envoyer et extraire des messages protégés contre l'intégrité vers et depuis des files d'attente gérées par deux gestionnaires de files d'attente différents. Les deux gestionnaires de files d'attente peuvent être exécutés sur le même système z/OS ou sur des systèmes z/OS différents, ou un gestionnaire de files d'attente peut être exécuté sur un système réparti exécutant Advanced Message Security.

Les exemples de gestionnaires de files d'attente et de files d'attente sont les suivants:

```
BNK6        - Sending queue manager
BNK7        - Recipient queue manager
FIN.XFER.Q7 - Remote queue on BNK6
FIN.RCPT.Q7 - Local queue on BNK7
```

Remarque: Dans cet exemple, BNK6 et BNK7 sont des gestionnaires de files d'attente s'exécutant sur des systèmes z/OS différents.

Ces utilisateurs sont utilisés:

```
WMQBNK6  - AMS task user on BNK6
WMQBNK7  - AMStask user on BNK7
TELLER5  - Sending user on BNK6
FINADM2  - Recipient user on BNK7
```

Les étapes de configuration de ce scénario sont les suivantes:

Créer les certificats d'utilisateur

Dans cet exemple, un seul certificat d'utilisateur est requis. Il s'agit du certificat de l'utilisateur émetteur qui est nécessaire pour signer le message protégé contre l'intégrité. L'utilisateur émetteur est'TELLER5'.

Le certificat de l'autorité de certification est également requis.Le certificat de l'autorité de certification est le certificat de l'autorité qui a émis le certificat de l'utilisateur. Il peut s'agir d'une chaîne de certificats. Si tel est le cas, tous les certificats de la chaîne sont requis dans le fichier de clés de l'utilisateur de la tâche Advanced Message Security , dans ce cas l'utilisateur WMQBNK7.

Un certificat d'autorité de certification peut être créé à l'aide de la commande RACDCERT RACF . Ce certificat est utilisé pour émettre des certificats d'utilisateur. Exemple :

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Cette commande RACDCERT crée un certificat d'autorité de certification qui peut ensuite être utilisé pour émettre un certificat d'utilisateur pour l'utilisateur'TELLER5'. Exemple :

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

Votre installation comporte des procédures pour choisir ou créer un certificat d'autorité de certification, ainsi que des procédures pour émettre des certificats et les distribuer aux systèmes appropriés.

Lors de l'exportation et de l'importation de ces certificats, Advanced Message Security requiert:

- Certificat de l'autorité de certification (chaîne).
- Certificat d'utilisateur émetteur et sa clé privée.

Si vous utilisez RACF, la commande RACDCERT EXPORT peut être utilisée pour exporter des certificats vers un fichier et la commande RACDCERT ADD peut être utilisée pour importer des certificats à partir du fichier. Pour plus d'informations sur ces commandes et d'autres commandes RACDCERT, voir RACDCERT (Manage RACF digital certificates) dans le manuel *z/OS: Security Server RACF Command Language Reference*.

Dans ce cas, les certificats sont requis sur le système z/OS exécutant le gestionnaire de files d'attente BNK6 et BNK7.

Dans cet exemple, le certificat d'envoi doit être importé sur le système z/OS exécutant BNK6, et le certificat de l'autorité de certification doit être importé sur le système z/OS exécutant BNK7. Lorsque les certificats ont été importés, le certificat utilisateur requiert l'attribut TRUST. La commande RACDCERT ALTER peut être utilisée pour ajouter l'attribut TRUST au certificat. Par exemple, sous BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
```

Connexion de certificats à des fichiers de clés pertinents

Une fois que les certificats requis ont été créés ou importés et définis comme étant de confiance, ils doivent être connectés aux fichiers de clés utilisateur appropriés sur le système z/OS exécutant BNK6 et BNK7.

Pour créer les fichiers de clés, utilisez la commande RACDCERT ADDRING, sur BNK6:

```
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur émetteur sur BNK6. Notez que le nom de fichier de clés drq.ams.keyring est obligatoire et qu'il est sensible à la casse.

Sur BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur de la tâche Advanced Message Security sur BNK7. Aucun fichier de clés utilisateur n'est requis pour'TELLER5'sur BNK7.

Une fois les fichiers de clés créés, les certificats appropriés peuvent être connectés.

Sur BNK6:

```
RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Sur BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))
```

Le certificat d'utilisateur émetteur doit être connecté en tant que DEFAULT. Si l'utilisateur émetteur possède plusieurs certificats dans son fichier drq.ams.keyring, le certificat par défaut est utilisé à des fins de signature.

La création et la modification de certificats ne sont pas reconnues par Advanced Message Security tant que le gestionnaire de files d'attente n'est pas arrêté et redémarré ou que la commande z/OS **MODIFY** n'est pas utilisée pour actualiser la configuration de certificat Advanced Message Security . Exemple :

Sur BNK6:

```
F BNK6AMSM,REFRESH,KEYRING
```

Sur BNK7:

```
F BNK7AMSM,REFRESH,KEYRING
```

Création des règles Advanced Message Security

Dans cet exemple, les messages protégés contre l'intégrité sont placés dans la file d'attente éloignée FIN.XFER.Q7 sur BNK6 par une application s'exécutant en tant qu'utilisateur'TELLER5'et extraite de la file d'attente locale FIN.RCPT.Q7 sur BNK7 par une application s'exécutant en tant qu'utilisateur'FINADM2', deux règles Advanced Message Security sont requises.

Les règles Advanced Message Security sont créées à l'aide de l'utilitaire CSQ0UTIL documenté à l'adresse The message security policy utility (CSQ0UTIL).

Utilisez l'utilitaire CSQ0UTIL pour exécuter la commande suivante afin de définir une règle d'intégrité pour la file d'attente éloignée sur BNK6:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK6. Le nom de la règle et la file d'attente associée sont FIN.XFER.Q7. L'algorithme utilisé pour générer la signature de l'expéditeur est MD5et le nom distinctif (DN) de l'utilisateur émetteur est'CN=Teller5,O=BCO,C=US'.

Utilisez également l'utilitaire CSQ0UTIL pour exécuter la commande suivante afin de définir une règle d'intégrité pour la file d'attente locale sur BNK7:

```
setmqspl -m BNK7 -p FIN.RCPT.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK7. Le nom de la règle et la file d'attente associée sont FIN.RCPT.Q7. L'algorithme attendu pour la signature de l'expéditeur est MD5et le nom distinctif (DN) de l'utilisateur émetteur doit être'CN=Teller5,O=BCO,C=US'.

Après avoir défini les deux règles, redémarrez les gestionnaires de files d'attente BNK6 et BNK7 ou utilisez la commande z/OS **MODIFY** pour actualiser les configurations de règles Advanced Message Security . Exemple :

Sur BNK6:

```
F BNK6AMSM,REFRESH,POLICY
```

Sur BNK7:

```
F BNK7AMSM,REFRESH,POLICY
```

z/OS *Mise en file d'attente à distance des messages protégés par la confidentialité sur z/OS*

Cet exemple détaille les règles et les certificats Advanced Message Security nécessaires pour envoyer et extraire des messages protégés par la confidentialité vers et depuis des files d'attente gérées par deux gestionnaires de files d'attente différents. Les deux gestionnaires de files d'attente peuvent être exécutés sur le même système z/OS ou sur des systèmes z/OS différents, ou un gestionnaire de files d'attente peut être exécuté sur un système réparti exécutant Advanced Message Security.

Les exemples de gestionnaires de files d'attente et de files d'attente sont les suivants:

```
BNK6        - Sending queue manager
BNK7        - Recipient queue manager
FIN.XFER.Q7 - Remote queue on BNK6
FIN.RCPT.Q7 - Local queue on BNK7
```

Remarque: dans cet exemple, BNK6 et BNK7 sont des gestionnaires de files d'attente s'exécutant sur des systèmes z/OS différents portant le même nom.

Ces utilisateurs sont utilisés:

```
WMQBNK6  - AMS task user on BNK6
WMQBNK7  - AMS task user on BNK7
TELLER5  - Sending user on BNK6
FINADM2  - Recipient user on BNK7
```

Les étapes de configuration de ce scénario sont les suivantes:

Créer les certificats d'utilisateur

Dans cet exemple, deux certificats d'utilisateur sont requis. Il s'agit du certificat de l'utilisateur émetteur qui est nécessaire pour signer les messages et du certificat de l'utilisateur destinataire qui est nécessaire pour chiffrer et déchiffrer les données du message. L'utilisateur émetteur est'TELLER5'et l'utilisateur destinataire est'FINADM2'.

Le certificat de l'autorité de certification est également requis.Le certificat de l'autorité de certification est le certificat de l'autorité qui a émis le certificat de l'utilisateur. Il peut s'agir d'une chaîne de certificats. Si tel est le cas, tous les certificats de la chaîne sont requis dans le fichier de clés de l'utilisateur de la tâche Advanced Message Security , dans ce cas l'utilisateur WMQBNK7.

Un certificat d'autorité de certification peut être créé à l'aide de la commande RACDCERT RACF . Ce certificat est utilisé pour émettre des certificats d'utilisateur. Exemple :

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Cette commande RACDCERT crée un certificat d'autorité de certification qui peut ensuite être utilisé pour émettre des certificats d'utilisateur pour les utilisateurs'TELLER5'et'FINADM2'. Exemple :

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

Votre installation comporte des procédures pour choisir ou créer un certificat d'autorité de certification, ainsi que des procédures pour émettre des certificats et les distribuer aux systèmes appropriés.

Lors de l'exportation et de l'importation de ces certificats, Advanced Message Security requiert:

- Certificat de l'autorité de certification (chaîne).
- Certificat d'utilisateur émetteur et sa clé privée.
- Certificat d'utilisateur du destinataire et sa clé privée.

Si vous utilisez RACF, la commande RACDCERT EXPORT peut être utilisée pour exporter des certificats vers un fichier et la commande RACDCERT ADD peut être utilisée pour importer des certificats à partir du fichier.

Pour plus d'informations sur ces commandes et d'autres commandes RACDCERT, voir RACDCERT (Manage RACF digital certificates) dans le document *z/OS: Security Server RACF Command Language Reference*.

Dans ce cas, les certificats sont requis sur le système z/OS exécutant le gestionnaire de files d'attente BNK6 et BNK7.

Dans cet exemple, les certificats d'envoi et de destinataire doivent être importés sur le système z/OS exécutant BNK6, et les certificats d'autorité de certification et de destinataire doivent être importés sur le système z/OS exécutant BNK7. Lorsque les certificats ont été importés, les certificats d'utilisateur requièrent l'attribut TRUST. La commande RACDCERT ALTER peut être utilisée pour ajouter l'attribut TRUST au certificat. Exemple :

Sur BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Sur BNK7:

```
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Connexion de certificats à des fichiers de clés pertinents

Une fois que les certificats requis ont été créés ou importés et définis comme certificats de confiance, ils doivent être connectés aux fichiers de clés utilisateur appropriés sur les systèmes z/OS exécutant BNK6 et BNK7.

Pour créer les fichiers de clés, utilisez la commande RACDCERT ADDRING:

Sur BNK6:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur de la tâche Advanced Message Security et un fichier de clés pour l'utilisateur émetteur sur BNK6. Notez que le nom de fichier de clés drq.ams.keyring est obligatoire et qu'il est sensible à la casse.

Sur BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Cela crée un fichier de clés pour l'utilisateur de la tâche Advanced Message Security et un fichier de clés pour l'utilisateur destinataire sur BNK7.

Une fois les fichiers de clés créés, les certificats appropriés peuvent être connectés.

Sur BNK6:

```
RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Sur BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Les certificats d'utilisateur d'envoi et de destinataire doivent être connectés sous la forme DEFAULT. Si l'un des utilisateurs possède plusieurs certificats dans son fichier de clés drq.ams.keyring, le certificat par défaut est utilisé à des fins de signature et de chiffrement / déchiffrement.

Sur BNK6, le certificat de l'utilisateur destinataire doit également être connecté au fichier de clés de l'utilisateur de la tâche Advanced Message Security avec USAGE (SITE). En effet, la tâche Advanced Message Security a besoin de la clé publique du destinataire lors du chiffrement des données de message. USAGE (SITE) empêche la clé privée d'être accessible dans le fichier de clés.

La création et la modification de certificats ne sont pas reconnues par Advanced Message Security tant que le gestionnaire de files d'attente n'est pas arrêté et redémarré ou que la commande z/OS **MODIFY** n'est pas utilisée pour actualiser la configuration de certificat Advanced Message Security . Exemple :

Sur BNK6:

```
F BNK6AMSM,REFRESH,KEYRING
```

Sur BNK7:

```
F BNK7AMSM,REFRESH,KEYRING
```

Création des règles Advanced Message Security

Dans cet exemple, les messages protégés par la confidentialité sont placés dans la file d'attente éloignée FIN.XFER.Q7 sur BNK6 par une application s'exécutant en tant qu'utilisateur'TELLER5'et extraite de la file d'attente locale FIN.RCPT.Q7 sur BNK7 par une application s'exécutant en tant qu'utilisateur'FINADM2', deux règles Advanced Message Security sont requises.

Les règles Advanced Message Security sont créées à l'aide de l'utilitaire CSQ0UTIL documenté à l'adresse The message security policy utility (CSQ0UTIL).

Utilisez l'utilitaire CSQ0UTIL pour exécuter la commande suivante afin de définir une règle de confidentialité pour la file d'attente éloignée sur BNK6:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK6. Le nom de la règle et la file d'attente associée sont FIN.XFER.Q7. L'algorithme utilisé pour générer la signature de l'expéditeur est SHA1, le nom distinctif (DN) de l'utilisateur émetteur est'CN=Teller5,O=BCO,C=US'et l'utilisateur destinataire est'CN=FinAdm2,O=BCO,C=US'. L'algorithme utilisé pour chiffrer les données de message est 3DES.

Utilisez également l'utilitaire CSQ0UTIL pour exécuter la commande suivante afin de définir une règle de confidentialité pour la file d'attente locale sur BNK7:

```
setmqspl -m BNK7 -p FIN.RCPT.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

Dans cette règle, le gestionnaire de files d'attente est identifié comme BNK7. Le nom de la règle et la file d'attente associée sont FIN.RCPT.Q7. L'algorithme attendu pour la signature de l'expéditeur est SHA1, le nom distinctif (DN) de l'utilisateur émetteur est'CN=Teller5,O=BCO,C=US'et l'utilisateur destinataire est'CN=FinAdm2,O=BCO,C=US'. L'algorithme utilisé pour déchiffrer les données de message est 3DES.

Après avoir défini les deux règles, redémarrez les gestionnaires de files d'attente BNK6 et BNK7 ou utilisez la commande z/OS **MODIFY** pour actualiser la configuration des règles Advanced Message Security . Exemple :

Sur BNK6:

```
F BNK6AMSM,REFRESH,POLICY
```

Sur BNK7:

```
F BNK7AMSM,REFRESH,POLICY
```

Guide de démarrage rapide pour AMS avec les clients Java

Utilisez ce guide pour configurer rapidement Advanced Message Security afin de garantir la sécurité des messages pour les applications Java qui se connectent à l'aide de liaisons client. Lorsque vous l'aurez terminé, vous aurez créé un magasin de clés pour vérifier les identités utilisateur et défini des règles de signature / chiffrement pour votre gestionnaire de files d'attente.

Avant de commencer

Vérifiez que les composants appropriés sont installés, comme décrit dans le **Guide de démarrage rapide** (Windows ou UNIX).

1. Création d'un gestionnaire de files d'attente et d'une file d'attente

Pourquoi et quand exécuter cette tâche

Tous les exemples suivants utilisent une file d'attente nommée TEST.Q pour la transmission de messages entre les applications. Advanced Message Security utilise des intercepteurs pour signer et chiffrer les messages lorsqu'ils entrent dans l'infrastructure IBM MQ via l'interface IBM MQ standard. La configuration de base est effectuée dans IBM MQ et est configurée dans les étapes suivantes.

Procédure

1. Création d'un gestionnaire de files d'attente

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Démarrer le gestionnaire de files d'attente

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Créez et démarrez un programme d'écoute en entrant les commandes suivantes dans **runmqsc** pour le gestionnaire de files d'attente QM_VERIFY_AMS

   ```
   DEFINE LISTENER(AMS.LSTR) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR)
   START LISTENER(AMS.LSTR)
   ```

4. Créez un canal via lequel nos applications se connectent en entrant la commande suivante dans **runmqsc** pour le gestionnaire de files d'attente QM_VERIFY_AMS

   ```
   DEFINE CHANNEL(AMS.SVRCONN) CHLTYPE(SVRCONN)
   ```

5. Créez une file d'attente appelée TEST.Q en entrant la commande suivante dans **runmqsc** pour le gestionnaire de files d'attente QM_VERIFY_AMS

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Résultats

Si la procédure a abouti, la commande suivante entrée dans **runmqsc** affiche les détails relatifs à TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Création et autorisation d'utilisateurs

Pourquoi et quand exécuter cette tâche

Deux utilisateurs apparaissent dans ce scénario: alice, l'expéditeur, et bob, le destinataire. Pour utiliser la file d'attente d'application, ces utilisateurs doivent être autorisés à l'utiliser. De même, pour utiliser correctement les règles de protection définies dans ce scénario, ces utilisateurs doivent être autorisés à accéder à certaines files d'attente système. Pour plus d'informations sur la commande **setmqaut** , voir **setmqaut**.

Procédure

1. Créez les deux utilisateurs comme décrit dans le **Guide de démarrage rapide** (Windows ou UNIX) pour votre plateforme.
2. Autoriser les utilisateurs à se connecter au gestionnaire de files d'attente et à utiliser la file d'attente

   ```
   setmqaut -m QM_VERIFY_AMS -t qmgr -p alice -p bob +connect +inq
   setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p alice +put
   setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p bob +get +inq +browse
   ```

3. Vous devez également autoriser les deux utilisateurs à parcourir la file d'attente de règles système et à placer des messages dans la file d'attente d'erreurs.

   ```
   setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.POLICY.QUEUE -p alice -p bob +browse
   setmqaut -m QM_VERIFY_AMS -t queue -n SYSTEM.PROTECTION.ERROR.QUEUE -p alice -p bob +put
   ```

   Avertissement : IBM MQ optimise les performances en mettant en cache les règles de sorte que vous n'ayez pas à parcourir les enregistrements pour obtenir des détails sur les règles dans SYSTEM.PROTECTION.POLICY.QUEUE dans tous les cas.

   IBM MQ ne met pas en cache toutes les règles disponibles. S'il existe un nombre élevé de règles, IBM MQ met en cache un nombre limité de règles. Par conséquent, si le nombre de règles définies pour le gestionnaire de files d'attente est faible, il n'est pas nécessaire de fournir l'option de navigation à SYSTEM.PROTECTION.POLICY.QUEUE.

   Toutefois, vous devez accorder le droit de navigation à cette file d'attente, au cas où un nombre élevé de règles serait défini, ou si vous utilisez d'anciens clients. SYSTEM.PROTECTION.ERROR.QUEUE est utilisé pour placer les messages d'erreur générés par le code AMS. Le droit d'insertion sur cette file d'attente est vérifié uniquement lorsque vous tentez d'insérer un message d'erreur dans la file d'attente. Votre droit d'insertion sur la file d'attente n'est pas vérifié lorsque vous tentez d'insérer ou d'extraire un message d'une file d'attente protégée AMS.

Résultats

Les utilisateurs sont maintenant créés et les droits requis leur sont accordés.

Que faire ensuite

Pour vérifier si les étapes ont été effectuées correctement, utilisez les exemples JmsProducer et JmsConsumer comme décrit dans la section «7. Test de la configuration», à la page 618.

3. Création d'une base de données de clés et de certificats

Pourquoi et quand exécuter cette tâche

Pour chiffrer le message à l'intercepteur, la clé publique des utilisateurs qui l'envoient est requise. Par conséquent, la base de données de clés des identités utilisateur mappées aux clés publiques et privées doit être créée. Dans le système réel, où les utilisateurs et les applications sont dispersés sur plusieurs ordinateurs, chaque utilisateur dispose de son propre magasin de clés privé. De même, dans ce guide, nous créons des bases de données de clés pour `alice` et bob et nous partageons les certificats d'utilisateur entre eux.

Remarque : Dans ce guide, nous utilisons des exemples d'applications écrits en Java se connectant à l'aide de liaisons client. Si vous prévoyez d'utiliser des applications Java à l'aide de liaisons locales ou d'applications C, vous devez créer un magasin de clés CMS et des certificats à l'aide de la commande **runmqakm** . Cela est indiqué dans le **Guide de démarrage rapide** (Windows ou UNIX).

Procédure

1. Créez un répertoire dans lequel créer votre magasin de clés, par exemple `/home/alice/.mqs`. Vous souhaiterez peut-être le créer dans le même répertoire que celui utilisé par le **Guide de démarrage rapide** (Windows ou UNIX) pour votre plateforme.

   Remarque : Ce répertoire est appelé *keystore-dir* dans les étapes suivantes
2. Création d'un fichier de clés et d'un certificat identifiant l'utilisateur `alice` à utiliser dans le chiffrement

   Remarque : La commande **keytool** fait partie de l'environnement d'exécution Java.

   ```
   keytool -genkey -alias Alice_Java_Cert -keyalg RSA -keystore keystore-dir/keystore.jks
   -storepass passw0rd
   -dname "CN=alice, O=IBM, C=GB" -keypass passw0rd
   ```

   Remarque :

   - Si votre *keystore-dir* contient des espaces, vous devez placer des guillemets autour du nom complet de votre magasin de clés
   - Il est conseillé d'utiliser un mot de passe fiable pour sécuriser le magasin de clés.
   - Pour les besoins de ce guide, nous utilisons un certificat autosigné qui peut être créé sans utiliser d'autorité de certification. Pour les systèmes de production, il est conseillé de ne pas utiliser de certificats autosignés, mais de s'appuyer sur des certificats signés par une autorité de certification.
   - Le paramètre **alias** indique le nom du certificat, que les intercepteurs recherchent pour recevoir les informations nécessaires.
   - Le paramètre **dname** spécifie les détails du **nom distinctif** (DN), qui doit être unique pour chaque utilisateur.
3. Sous UNIX, vérifiez que le magasin de clés est lisible

   ```
   chmod +r keystore-dir/keystore.jks
   ```
4. Répétez l' step1-4 pour l'utilisateur bob

Résultats

Les deux utilisateurs `alice` et bob possèdent chacun un certificat autosigné.

4. Création de keystore.conf

Pourquoi et quand exécuter cette tâche

Vous devez pointer les intercepteurs Advanced Message Security vers le répertoire dans lequel se trouvent les bases de données de clés et les certificats. Cette opération est effectuée via le fichier

keystore.conf , qui contient ces informations sous forme de texte en clair. Chaque utilisateur doit disposer d'un fichier keystore.conf distinct. Cette étape doit être effectuée pour alice et bob.

Exemple

Pour ce scénario, le contenu de keystore.conf for alice est le suivant:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Alice_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = passw0rd
JKS.key_pass = passw0rd
JKS.provider = IBMJCE
```

Pour ce scénario, le contenu de keystore.conf for bob est le suivant:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Bob_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = passw0rd
JKS.key_pass = passw0rd
JKS.provider = IBMJCE
```

Remarque :

- Le chemin d'accès au fichier de clés doit être fourni sans inclure l'extension de fichier.
- Si vous disposez déjà d'un fichier keystore.conf car vous avez suivi les instructions du guide de démarrage rapide (Windows ou UNIX), vous pouvez éditer le fichier existant pour ajouter ces lignes.
- Pour plus d'informations, voir «Structure du fichier de configuration du magasin de clés (keystore.conf) pour AMS», à la page 627.

5. Partage de certificats

Pourquoi et quand exécuter cette tâche

Partagez les certificats entre les deux magasins de clés afin que chaque utilisateur puisse identifier l'autre. Cette opération est effectuée en extrayant le certificat de chaque utilisateur et en l'important dans le magasin de clés de l'autre utilisateur.

Remarque : Les termes *extraire* et *exporter* sont utilisés différemment par les différents outils de certificat. Par exemple, l'outil de commande IBM GSKit **strmqikm** (ikeyman) distingue que vous *extrayez* des certificats (clés publiques) et que vous *exportez* des clés privées. Cette distinction est extrêmement importante pour les outils qui offrent les deux options, car l'utilisation de *export* par erreur compromettrait complètement votre application en transmettant sa clé privée. Etant donné que la distinction est si importante, la documentation IBM MQ s'efforce d'utiliser ces termes de manière cohérente. Toutefois, l'outil de clé Java fournit une option de ligne de commande appelée *exportcert* qui extrait uniquement la clé publique. Pour ces raisons, la procédure suivante fait référence à l' *extraction de certificats* à l'aide de l'option *exportcert* .

Procédure

1. Extrayez le certificat identifiant alice.

   ```
   keytool -exportcert -keystore alice-keystore-dir/keystore.jks -storepass passw0rd
   -alias Alice_Java_Cert -file alice-keystore-dir/Alice_Java_Cert.cer
   ```

2. Importez le certificat identifiant alice dans le magasin de clés que bob utilisera. Lorsque vous y êtes invité, indiquez que vous ferez confiance à ce certificat.

   ```
   keytool -importcert -file alice-keystore-dir/Alice_Java_Cert.cer -alias Alice_Java_Cert
   -keystore bob-keystore-dir/keystore.jks -storepass passw0rd
   ```

3. Répétez les étapes pour bob

Résultats

Les deux utilisateurs `alice` et bob sont désormais en mesure de s'identifier mutuellement en ayant créé et partagé des certificats autosignés.

Que faire ensuite

Vérifiez qu'un certificat se trouve dans le magasin de clés en exécutant les commandes suivantes qui impriment ses détails:

```
keytool -list -keystore bob-keystore-dir/keystore.jks -storepass passw0rd -alias Alice_Java_Cert
keytool -list -keystore alice-keystore-dir/keystore.jks -storepass passw0rd -alias Bob_Java_Cert
```

6. Définition de la règle de file d'attente

Pourquoi et quand exécuter cette tâche

Avec le gestionnaire de files d'attente créé et les intercepteurs préparés pour intercepter les messages et les clés de chiffrement d'accès, nous pouvons commencer à définir des règles de protection sur `QM_VERIFY_AMS` à l'aide de la commande `setmqspl` . Pour plus d'informations sur cette commande, voir setmqspl . Chaque nom de règle doit être identique au nom de la file d'attente à laquelle il doit être appliqué.

Exemple

Voici un exemple de règle définie dans la file d'attente `TEST.Q` , signée par l'utilisateur `alice` à l'aide de l'algorithme SHA1 et chiffrée à l'aide de l'algorithme AES 256 bits pour l'utilisateur bob:

```
setmqspl -m QM_VERIFY_AMS -p TEST.Q -s SHA1 -a "CN=alice,O=IBM,C=GB" -e AES256 -r
"CN=bob,O=IBM,C=GB"
```

Remarque : Les noms distinctifs correspondent exactement à ceux spécifiés dans le certificat de l'utilisateur respectif de la base de données de clés.

Que faire ensuite

Pour vérifier la règle que vous avez définie, exécutez la commande suivante:

```
dspmqspl -m QM_VERIFY_AMS
```

Pour imprimer les détails de la règle sous la forme d'un ensemble de commandes `setmqspl` , utilisez l'indicateur `-export` . Cela permet de stocker des règles déjà définies:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Test de la configuration

Avant de commencer

Vérifiez que les fichiers de règles JCE sans restriction sont installés dans la version de Java que vous utilisez.

Remarque : La version de Java fournie dans l'installation IBM MQ contient déjà ces fichiers de règles. Il se trouve dans *MQ_INSTALLATION_PATH*`/java/bin`.

Pourquoi et quand exécuter cette tâche

En exécutant différents programmes sous différents utilisateurs, vous pouvez vérifier si l'application a été correctement configurée. Voir le **Guide de démarrage rapide** (Windows ou UNIX) pour votre plateforme, pour plus de détails sur l'exécution de programmes sous différents utilisateurs.

Procédure

1. Pour exécuter ces modèles d'application JMS , utilisez le paramètre CLASSPATH pour votre plateforme, comme indiqué dans la rubrique Variables d'environnement utilisées par IBM MQ classes for JMS pour vous assurer que le répertoire des exemples est inclus.
2. En tant qu'utilisateur `alice`, placez un message à l'aide d'un exemple d'application, en vous connectant en tant que client:

   ```
   java JmsProducer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

3. En tant qu'utilisateur bob, obtenez un message à l'aide d'un exemple d'application, en vous connectant en tant que client:

   ```
   java JmsConsumer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

Résultats

Si l'application a été correctement configurée pour les deux utilisateurs, le message de l'utilisateur `alice` s'affiche lorsque bob exécute l'application d'obtention.

Protection des files d'attente éloignées

Pour protéger complètement les files d'attente éloignées, des règles doivent être définies sur la file d'attente éloignée et la file d'attente locale à laquelle les messages sont transmis.

Lorsqu'un message est inséré dans une file d'attente éloignée, Advanced Message Security intercepte l'opération et traite le message conformément à un ensemble de règles pour la file d'attente éloignée. Par exemple, pour une règle de chiffrement, le message est chiffré avant d'être transmis à IBM MQ pour le gérer. Une fois que Advanced Message Security a traité le message inséré dans une file d'attente éloignée, IBM MQ le place dans la file d'attente de transmission associée et le transmet au gestionnaire de files d'attente cible et à la file d'attente cible.

Lorsqu'une opération GET est effectuée sur la file d'attente locale, Advanced Message Security tente de décoder le message en fonction de l'ensemble de règles de la file d'attente locale. Pour que l'opération aboutisse, la règle utilisée pour déchiffrer le message doit être identique à celle utilisée pour le chiffrer. Toute différence provoquera le rejet du message.

Si, pour une raison quelconque, les deux règles ne peuvent pas être définies en même temps, une prise en charge du déploiement par étapes est fournie. La règle peut être définie sur une file d'attente locale avec l'indicateur de tolérance activé, qui indique qu'une règle associée à une file d'attente peut être ignorée lorsqu'une tentative d'extraction d'un message de la file d'attente implique un message pour lequel la règle de sécurité n'est pas définie. Dans ce cas, GET tente de déchiffrer le message, mais autorise la distribution de messages non chiffrés. De cette manière, les règles des files d'attente éloignées peuvent être définies une fois que les files d'attente locales ont été protégées (et testées).

A faire : Supprimez l'indicateur de tolérance une fois le déploiement de Advanced Message Security terminé.

Référence associée

setmqspl (définition de la règle de sécurité)

Routage des messages protégés à l'aide de IBM Integration Bus

Advanced Message Security peut protéger les messages dans une infrastructure où IBM Integration Busou WebSphere Message Broker 8.0.0.1 (ou version ultérieure) est installé. Vous devez comprendre la nature des deux produits avant d'appliquer la sécurité dans l'environnement IBM Integration Bus .

Pourquoi et quand exécuter cette tâche

Advanced Message Security fournit une sécurité de bout en bout de la charge de message. Cela signifie que seules les parties spécifiées comme expéditeurs et destinataires valides d'un message sont capables de le produire ou de le recevoir. Cela implique que pour sécuriser les messages transitant par IBM Integration Bus, vous pouvez autoriser IBM Integration Bus à traiter les messages sans connaître leur contenu (Scénario 1) ou en faire un utilisateur autorisé à recevoir et à envoyer des messages (Scénario 2).

Scénario 1- Integration Bus ne peut pas voir le contenu des messages

Avant de commencer

IBM Integration Bus doit être connecté à un gestionnaire de files d'attente existant. Remplacez *QMgrName* par ce nom de gestionnaire de files d'attente existant dans les commandes qui suivent.

Pourquoi et quand exécuter cette tâche

Dans ce scénario, Alice place un message protégé dans une file d'attente d'entrée `QIN`. En fonction de la propriété de message `routeTo`, le message est acheminé vers *Bob's* (`QBOB`),[1](`QCECIL`) ou la file d'attente par défaut (`QDEF`). Le routage est possible car Advanced Message Security protège uniquement la charge de message et non ses en-têtes et propriétés qui restent non protégés et peuvent être lus par IBM Integration Bus. Advanced Message Security est utilisé uniquement par *alice*, *bob* et *cecil*. Il n'est pas nécessaire de l'installer ou de le configurer pour IBM Integration Bus.

IBM Integration Bus reçoit le message protégé de la file d'attente d'alias non protégée afin d'éviter toute tentative de déchiffrement du message. S'il devait utiliser directement la file d'attente protégée, le message serait placé dans la file d'attente DEAD LETTER comme impossible à déchiffrer. Le message est acheminé par IBM Integration Bus et arrive dans la file d'attente cible sans modification. Par conséquent, il est toujours signé par l'auteur d'origine (*bob* et *cecil* n'acceptent que les messages envoyés par *alice*) et protégé comme précédemment (seuls *bob* et *cecil* peuvent le lire). IBM Integration Bus place le message acheminé dans un alias non protégé. Les destinataires extraient le message d'une file d'attente en sortie protégée où AMS déchiffre le message de manière transparente.

Procédure

1. Configurez *alice*, *bob* et *cecil* pour utiliser Advanced Message Security comme décrit dans le **Guide de démarrage rapide** (Windows ou UNIX).

   Vérifiez que les étapes suivantes sont effectuées:

   - Création et autorisation d'utilisateurs
   - Création de la base de données de clés et des certificats
   - Création de keystore.conf
2. Fournissez le certificat *alice* à *bob* et *cecil*, de sorte que *alice* puisse être identifié par eux lors de la vérification des signatures numériques sur les messages.

   Pour ce faire, extrayez le certificat identifiant *alice* dans un fichier externe, puis ajoutez le certificat extrait aux magasins de clés *Bob's* et *Cecil's* . Il est important d'utiliser la méthode décrite dans la tâche 5 de **. Partage de certificats** dans le **Guide de démarrage rapide** (Windows ou UNIX).
3. Fournissez les certificats *bob* et *cecil* à *alice*, de sorte que *alice* puisse envoyer des messages chiffrés pour *bob* et *cecil*.

   Effectuez cette opération à l'aide de la méthode spécifiée à l'étape précédente.
4. Sur votre gestionnaire de files d'attente, définissez des files d'attente locales appelées `QIN`, `QBOB`, `QCECIL` et `QDEF`.

   ```
   DEFINE QLOCAL(QIN)
   ```

[1] Cecil's

5. Définissez la règle de sécurité pour la file d'attente QIN sur une configuration éligible. Utilisez la configuration identique pour les files d'attente QBOB, QCECIL et QDEF .

```
setmqspl -m QMgrName -p QIN -s SHA1 -a "CN=alice,O=IBM,C=GB"
-e AES256 -r "CN=bob,O=IBM,C=GB" -r "CN=cecil,O=IBM,C=GB"
```

   Ce scénario suppose la règle de sécurité dans laquelle *alice* est le seul expéditeur autorisé et *bob* et *cecil* sont les destinataires.
6. Définissez des files d'attente alias AIN, ABOB et ACECIL référençant des files d'attente locales QIN, QBOB et QCECIL respectivement.

```
DEFINE QALIAS(AIN) TARGET(QIN)
```

7. Vérifiez que la configuration de sécurité pour les alias spécifiés à l'étape précédente n'est pas présente ; sinon, définissez sa règle sur NONE.

```
dspmqspl -m QMgrName -p AIN
```

8. Dans IBM Integration Bus , créez un flux de messages pour acheminer les messages arrivant dans la file d'attente alias AIN vers le noeud BOB, CECIL ou DEF en fonction de la propriété routeTo du message. Pour ce faire, procédez comme suit :
   a) Créez un noeud MQInput appelé IN et affectez l'alias AIN comme nom de file d'attente.
   b) Créez des noeuds MQOutput appelés BOB, CECIL et DEFet affectez des files d'attente alias ABOB, ACECIL et ADEF comme noms de file d'attente respectifs.
   c) Créez un noeud de route et appelez-le TEST.
   d) Connectez le noeud IN au terminal d'entrée du noeud TEST .
   e) Créez des terminaux de sortie bobet cecil pour le noeud TEST .
   f) Connectez le terminal de sortie bob au noeud BOB .
   g) Connectez le terminal de sortie cecil au noeud CECIL .
   h) Connectez le noeud DEF au terminal de sortie par défaut.
   i) Appliquez les règles suivantes:

```
$Root/MQRFH2/usr/routeTo/text()="bob"
$Root/MQRFH2/usr/routeTo/text()="cecil"
```

9. Déployez le flux de messages dans le composant d'exécution IBM Integration Bus .
10. L'exécution en tant qu'utilisateur Alice a inséré un message qui contient également une propriété de message appelée routeTo avec la valeur bob ou cecil. L'exécution du modèle d'application **amqsstm** vous permet d'effectuer cette opération.

```
Sample AMQSSTMA start
target queue is TEST.Q
Enter property name
routeTo
Enter property value
bob
Enter property name

Enter message text
My Message to Bob
Sample AMQSSTMA end
```

11. L'exécution en tant qu'utilisateur *bob* extrait le message de la file d'attente QBOB à l'aide du modèle d'application **amqsget**.

Résultats

Lorsque *alice* place un message dans la file d'attente `QIN` , le message est protégé. Il est extrait sous forme protégée par IBM Integration Bus à partir de la file d'attente d'alias `AIN` . IBM Integration Bus décide où acheminer le message en lisant la propriété `routeTo` qui est, comme toutes les propriétés, non chiffrée. IBM Integration Bus place le message sur l'alias non protégé approprié en évitant sa protection supplémentaire. Lorsqu'il est reçu par *bob* ou *cecil* de la file d'attente, le message est déchiffré et la signature numérique est vérifiée.

Scénario 2- Integration Bus peut voir le contenu des messages

Pourquoi et quand exécuter cette tâche

Dans ce scénario, un groupe d'individus est autorisé à envoyer des messages à IBM Integration Bus. Un autre groupe est autorisé à recevoir les messages créés par IBM Integration Bus. La transmission entre les parties et IBM Integration Bus ne peut pas être espionner.

N'oubliez pas que IBM Integration Bus lit les règles de protection et les certificats uniquement lorsqu'une file d'attente est ouverte. Vous devez donc recharger le groupe d'exécution après avoir apporté des mises à jour aux règles de protection pour que les modifications soient prises en compte.

```
mqsireload execution-group-name
```

Si IBM Integration Bus est considéré comme une partie autorisée à lire ou à signer la charge de message, vous devez configurer Advanced Message Security pour l'utilisateur qui démarre le service IBM Integration Bus . Sachez que ce n'est pas nécessairement le même utilisateur qui insère / extrait les messages dans les files d'attente, ni l'utilisateur qui crée et déploie les applications IBM Integration Bus .

Procédure

1. Configurez *alice*, *bob*, *cecil* et *dave* et l'utilisateur du service IBM Integration Bus pour utiliser Advanced Message Security comme décrit dans le **Guide de démarrage rapide** (Windows ou UNIX).

   Vérifiez que les étapes suivantes sont effectuées:

   - Création et autorisation d'utilisateurs
   - Création de la base de données de clés et des certificats
   - Création de keystore.conf
2. Fournissez les certificats *alice*, *bob*, *cecil* et *dave* à l'utilisateur du service IBM Integration Bus .

   Pour ce faire, extrayez chacun des certificats identifiant *alice*, *bob*, *cecil* et *dave* dans des fichiers externes, puis ajoutez les certificats extraits au magasin de clés IBM Integration Bus . Il est important d'utiliser la méthode décrite dans la tâche 5 de **. Partage de certificats** dans le **Guide de démarrage rapide** (Windows ou UNIX).
3. Fournissez le certificat de l'utilisateur de service IBM Integration Bus à *alice*, *bob*, *cecil* et *dave*.

   Effectuez cette opération à l'aide de la méthode spécifiée à l'étape précédente.

   Remarque : *Alice* et *bob* ont besoin du certificat de l'utilisateur du service IBM Integration Bus pour chiffrer correctement les messages. L'utilisateur du service IBM Integration Bus a besoin des certificats *alice* et *bob* pour vérifier les auteurs des messages. L'utilisateur du service IBM Integration Bus a besoin des certificats *cecil* et *dave* pour chiffrer les messages qui lui sont destinés. *cecil* et *dave* ont besoin du certificat de l'utilisateur du service IBM Integration Bus pour vérifier si le message provient de IBM Integration Bus.
4. Définissez une file d'attente locale nommée `IN` et définissez la règle de sécurité avec *alice* et *bob* spécifiés comme auteurs, ainsi que l'utilisateur de service pour le IBM Integration Bus spécifié comme destinataire:

   ```
   setmqspl -m QMgrName -p IN -s MD5 -a "CN=alice,O=IBM,C=GB" -a "CN=bob,O=IBM,C=GB"
   -e AES256 -r "CN=broker,O=IBM,C=GB"
   ```

5. Définissez une file d'attente locale nommée OUTet définissez la règle de sécurité avec l'utilisateur de service pour IBM Integration Bus spécifié en tant qu'auteur et *cecil* et *dave* spécifié en tant que destinataires:

```
setmqspl -m QMgrName -p OUT -s MD5 -a "CN=broker,O=IBM,C=GB" -e AES256
-r "CN=cecil,O=IBM,C=GB" -r "CN=dave,O=IBM,C=GB"
```

6. Dans IBM Integration Bus , créez un flux de messages avec un noeud `MQInput` et `MQOutput` . Configurez le noeud `MQInput` pour qu'il utilise la file d'attente `IN` et le noeud `MQOutput` pour qu'il utilise la file d'attente OUT .
7. Déployez le flux de messages dans le composant d'exécution IBM Integration Bus .
8. L'exécution en tant qu'utilisateur *alice* ou *bob* a inséré un message dans la file d'attente `IN` à l'aide du modèle d'application **amqsput**.
9. L'exécution en tant qu'utilisateur *cecil* ou *dave* extrait le message de la file d'attente OUT à l'aide du modèle d'application **amqsget**.

Résultats

Les messages envoyés par *alice* ou *bob* à la file d'attente d'entrée `IN` sont chiffrés, ce qui permet uniquement à IBM Integration Bus de le lire. IBM Integration Bus n'accepte que les messages de *alice* et *bob* et rejette les autres. Les messages acceptés sont traités de manière appropriée, puis signés et chiffrés avec les clés *cecil* et *dave* avant d'être placés dans la file d'attente de sortie OUT. Seuls *cecil* et *dave* sont capables de le lire, les messages non signés par IBM Integration Bus sont rejetés.

Utilisation de Advanced Message Security avec Managed File Transfer

Ce scénario explique comment configurer Advanced Message Security pour fournir la confidentialité des messages pour les données envoyées via un Managed File Transfer.

Avant de commencer

Vérifiez que le composant Advanced Message Security est installé sur l'installation IBM MQ hébergeant les files d'attente utilisées par Managed File Transfer que vous souhaitez protéger.

Si vos agents Managed File Transfer se connectent en mode liaisons, assurez-vous que le composant GSKit est également installé sur leur installation locale.

Pourquoi et quand exécuter cette tâche

Lorsque le transfert de données entre deux agents Managed File Transfer est interrompu, il est possible que des données confidentielles restent non protégées dans les files d'attente IBM MQ sous-jacentes utilisées pour gérer le transfert. Ce scénario explique comment configurer et utiliser Advanced Message Security pour protéger ces données dans les files d'attente Managed File Transfer .

Dans ce scénario, nous considérons une topologie simple comprenant une machine avec deux files d'attente Managed File Transfer et deux agents, `AGENT1` et `AGENT2`, partageant un seul gestionnaire de files d'attente, comme décrit dans le scénario Présentation du scénario. Les deux agents se connectent de la même manière, soit en mode liaisons, soit en mode client.

1. Création de certificats

Avant de commencer

Ce scénario utilise un modèle simple dans lequel un utilisateur `ftagent` d'un groupe `FTAGENTS` est utilisé pour exécuter les processus Managed File Transfer Agent . Si vous utilisez vos propres noms d'utilisateur et de groupe, modifiez les commandes en conséquence.

Pourquoi et quand exécuter cette tâche

Advanced Message Security utilise la cryptographie à clé publique pour signer et / ou chiffrer les messages dans les files d'attente protégées.

Remarque :

- Si vos agents Managed File Transfer s'exécutent en mode liaisons, les commandes que vous utilisez pour créer un magasin de clés CMS (Cryptographic Message Syntax) sont détaillées dans le **Guide de démarrage rapide** (Windows ou UNIX) pour votre plateforme.
- Si vos agents Managed File Transfer s'exécutent en mode client, les commandes dont vous aurez besoin pour créer un fichier de clés JKS (Java Keystore) sont détaillées dans «Guide de démarrage rapide pour AMS avec les clients Java», à la page 614.

Procédure

1. Créez un certificat autosigné pour identifier l'utilisateur `ftagent` comme indiqué dans le guide de démarrage rapide approprié.
   Utilisez un nom distinctif (DN) comme suit:

   ```
   CN=ftagent, OU=MFT, O=<organisation>, L=<location>, ST=<state>, C=<country>
   ```

2. Créez un fichier `keystore.conf` pour identifier l'emplacement du magasin de clés et le certificat qu'il contient, comme indiqué dans le guide de démarrage rapide approprié.

2. Configuration de la protection des messages

Pourquoi et quand exécuter cette tâche

Vous devez définir une règle de sécurité pour la file d'attente de données utilisée par AGENT2, à l'aide de la commande **setmqspl** . Dans ce scénario, le même utilisateur est utilisé pour démarrer les deux agents et, par conséquent, le nom distinctif du signataire et du récepteur sont identiques et correspondent au certificat que nous avons généré.

Procédure

1. Arrêtez les agents Managed File Transfer en vue de leur protection à l'aide de la commande **fteStopAgent** .
2. Créez une règle de sécurité pour protéger la file d'attente SYSTEM.FTE.DATA.AGENT2 .

   ```
   setmqspl -m hubQM -p SYSTEM.FTE.DATA.AGENT2 -s SHA1 -a "CN=ftagent, OU=MFT,
   O=<organisation>, L=<location>, ST=<state>, C=<country>"
   -e AES128 -r "CN=ftagent, OU=MFT, O=<organisation>, L=<location>, ST=<state>, C=<country>"
   ```

3. Vérifiez que l'utilisateur exécutant le processus Managed File Transfer Agent a accès à la file d'attente de la règle système et placez les messages dans la file d'attente d'erreurs.

   ```
   setmqaut -m hubQM -t queue -n SYSTEM.PROTECTION.POLICY.QUEUE -p ftagent +browse
   setmqaut -m hubQM -t queue -n SYSTEM.PROTECTION.ERROR.QUEUE -p ftagent +put
   ```

4. Redémarrez vos agents Managed File Transfer à l'aide de la commande **fteStartAgent** .
5. Confirmez que vos agents ont été redémarrés avec succès à l'aide de la commande **fteListAgents** et vérifiez qu'ils sont à l'état READY .

Résultats

Vous pouvez maintenant soumettre des transferts depuis AGENT1 vers AGENT2et le contenu du fichier sera transmis de manière sécurisée entre les deux agents.

Présentation de l'installation de Advanced Message Security

Installez le composant Advanced Message Security sur différentes plateformes.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur les procédures d'installation, voir Installation de Advanced Message Security sur plusieurs plateformes et Installation de Advanced Message Security sur z/OS.

Tâches associées

Désinstallation de Advanced Message Security

z/OS Audit sous z/OS

Advanced Message Security (AMS) for z/OS permet d'effectuer un audit facultatif des opérations effectuées par les applications sur les files d'attente protégées par des règles. Lorsque cette option est activée, les enregistrements d'audit IBM System Management Facility (SMF) sont générés pour la réussite et l'échec de ces opérations dans les files d'attente protégées par des règles. Les opérations auditées incluent MQPUT, MQPUT1et MQGET.

L'audit est désactivé par défaut, mais vous pouvez l'activer en configurant _AMS_SMF_TYPE et _AMS_SMF_AUDIT dans le fichier Language Environment ® _CEE_ENVFILE configuré pour l'espace adresse AMS . Pour plus d'informations, voir Création de procédures pour Advanced Message Security. La variable _AMS_SMF_TYPE est utilisée pour désigner le type d'enregistrement SMF et est un nombre compris entre 128 et 255. Un type d'enregistrement SMF 180 est habituel, mais n'est pas obligatoire. L'audit est désactivé en spécifiant la valeur 0. La variable _AMS_SMF_AUDIT détermine si les enregistrements d'audit sont créés pour les opérations ayant abouti, les opérations ayant échoué ou les deux. Les options d'audit peuvent également être modifiées de manière dynamique lorsque AMS est actif à l'aide des commandes de l'opérateur. Pour plus d'informations, voir Fonctionnement d' Advanced Message Security.

L'enregistrement SMF est défini à l'aide de sous-types, le sous-type 1 étant un événement d'audit général. L'enregistrement SMF contient toutes les données relatives à la demande en cours de traitement.

L'enregistrement SMF est mappé par la macro CSQ0KSMF (notez le zéro dans le nom de la macro), qui est fournie dans la bibliothèque cible SCSQMACS. Si vous écrivez des programmes de réduction de données pour les données SMF, vous pouvez inclure cette macro de mappage pour faciliter le développement et la personnalisation des routines de post-traitement SMF.

Dans les enregistrements SMF produits par Advanced Message Security for z/OS, les données sont organisées en sections. L'enregistrement se compose des éléments suivants:

- un en-tête SMF standard
- une extension d'en-tête définie par Advanced Message Security pour z/OS
- une section de produit
- une section de données

La section produit de l'enregistrement SMF est toujours présente dans les enregistrements produits par Advanced Message Security for z/OS. La section de données varie en fonction du sous-type. Actuellement, un sous-type est défini et par conséquent, une seule section de données est utilisée.

SMF est décrit dans le manuel z/OS System Management Facilities (SA22-7630). Les types d'enregistrement valides sont décrits dans le membre SMFPRMxx du fichier PARMLIB de votre système. Pour plus d'informations, voir la documentation SMF.

Générateur de rapport d'audit Advanced Message Security (CSQ0USMF)

Advanced Message Security for z/OS fournit un outil de génération de rapport d'audit appelé CSQ0USMF qui est fourni dans la bibliothèque d'installation SCSQAUTH. Un exemple de JCL permettant d'exécuter l'utilitaire CSQ0USMF appelé CSQ40RSM est fourni dans la bibliothèque d'installation SCSQPROC.

Avant d'exécuter l'utilitaire CSQ0USMF , les enregistrements SMF de type 180 doivent être vidés des fichiers SMF du système vers un fichier séquentiel. Par exemple, ce JCL vide les enregistrements SMF de type 180 d'un fichier SMF et les transfère vers un fichier cible:

```
//IFAUDUMP EXEC PGM=IFASMFDP
```

```
//INDD1 DD DSN=SYSn.MANn.syst,DISP=SHR
//OUTDD1 DD DSN=your.target.dataset,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
INDD(INDD1,OPTIONS(DUMP))
OUTDD(OUTDD1,TYPE(180))
/*
```

Vous devez vérifier les noms de fichier SMF réels utilisés par votre installation. Le fichier cible des enregistrements vidés doit avoir un format d'enregistrement VBS et une longueur d'enregistrement de 32760.

Remarque : Si des flux de consignation SMF sont utilisés, vous devez utiliser le programme IFASMFDL pour vider un flux de consignation dans un fichier séquentiel. Pour obtenir un exemple du JCL utilisé, voir Traitement des enregistrements SMF de type 116 .

Le fichier cible peut ensuite être utilisé comme entrée dans l'utilitaire CSQ0USMF pour générer un rapport d'audit AMS . Exemple :

```
//STEP1 EXEC PGM=CSQ0USMF,
// PARM=('/ -SMFTYPE 180 -M qmgr')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//    DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SMFIN DD DSN=your.target.dataset,DISP=SHR
//
```

Le programme CSQ0USMF accepte deux paramètres facultatifs, répertoriés dans Tableau 97, à la page 626:

Tableau 97. Paramètres facultatifs CSQ0USMF

Paramètre	Valeur	Description
SMFTYPE	nnn	Type d'enregistrement SMF applicable au rapport d'audit. Le programme CSQ0USMF utilise uniquement les enregistrements SMF qui correspondent à la valeur SMFTYPE lors de la génération du rapport. Si vous n'indiquez pas SMFTYPE, la valeur par défaut 180 est utilisée.
M	qmgr	Nom du gestionnaire de files d'attente IBM MQ applicable au rapport d'audit. Si vous ne spécifiez pas le paramètre -M, le rapport d'audit inclura tous les enregistrements d'audit pour tous les gestionnaires de files d'attente représentés dans le fichier SMFIN.

Utilisation de magasins de clés et de certificats

Pour fournir une protection cryptographique transparente aux applications IBM MQ , Advanced Message Security utilise le fichier de clés, dans lequel sont stockés les certificats de clé publique et une clé privée. Sous z/OS, un fichier de clés SAF est utilisé à la place d'un fichier de clés.

Dans Advanced Message Security, les utilisateurs et les applications sont représentés par des identités PKI (Public Key Infrastructure). Ce type d'identité est utilisé pour signer et chiffrer les messages. L'identité PKI est représentée par la zone **Nom distinctif (DN)** du sujet dans un certificat associé à des messages signés et chiffrés. Pour qu'un utilisateur ou une application puisse chiffrer ses messages, il

doit avoir accès au fichier de clés dans lequel sont stockés les certificats et les clés privées et publiques associées.

Sous Windows et UNIX , l'emplacement du magasin de clés est fourni dans le fichier de configuration du magasin de clés, qui est `keystore.conf` par défaut. Chaque utilisateur Advanced Message Security doit disposer du fichier de configuration de magasin de clés qui pointe vers un fichier de magasin de clés. Advanced Message Security accepte le format suivant des fichiers de clés: `.kdb`, `.jceks`, `.jks`.

L'emplacement par défaut du fichier `keystore.conf` est:

- IBM i UNIX Sous UNIX et IBM i: `$HOME/.mqs/keystore.conf`
- Windows Sous Windows : *%HOMEDRIVE%%HOMEPATH%*`\.mqs\keystore.conf`

  Remarque : Le chemin d'accès sous Windows peut et doit spécifier l'identificateur d'unité si plusieurs lettres d'unité sont disponibles.

Si vous utilisez un nom de fichier de clés et un emplacement spécifiés, vous devez utiliser les commandes suivantes:

- Pour Java: `java -D`*`MQS_KEYSTORE_CONF=path/filename app_name`*
- Pour le client et le serveur C:
  - Sous UNIX and Linux : `export MQS_KEYSTORE_CONF=`*`path/filename`*
  - Sous Windows : `set MQS_KEYSTORE_CONF=`*`path\filename`*

Concepts associés

«Noms distinctifs d'expéditeur dans AMS», à la page 654
Les noms distinctifs d'expéditeur identifient les utilisateurs autorisés à placer des messages dans une file d'attente. Un expéditeur utilise son certificat pour signer un message, avant de le placer dans une file d'attente.

«Noms distinctifs des destinataires dans AMS», à la page 656
Les noms distinctifs des destinataires identifient les utilisateurs qui sont autorisés à extraire des messages d'une file d'attente.

Structure du fichier de configuration du magasin de clés (keystore.conf) pour AMS

Le fichier de configuration du magasin de clés (`keystore.conf`) pointe Advanced Message Security vers l'emplacement du magasin de clés approprié.

Chacun des types de fichier de configuration suivants possède un préfixe:

système moniteur conversationnel
Certificate Management System, les entrées de configuration sont préfixées avec: `cms.`

PKCS#11
Public Key Cryptography Standard #11, les entrées de configuration sont préfixées avec: `pkcs11.`

IBM i **marque d"erreur d"impression**
Format Privacy Enhanced Mail, les entrées de configuration sont préfixées avec: `pem.`

JKS
Java KeyStore, les entrées de configuration sont préfixées avec: `jks.`

JCEKS
Java Chiffrement cryptographique KeyStore, les entrées de configuration sont préfixées avec: `jceks.`

z/OS V 9.1.0 MQ Adv. VUE **JCERACFKS**
Java Cryptographic Encryption RACF keyring KeyStore, les entrées de configuration sont préfixées avec: `jceracfks`.

Important : A partir de IBM MQ 9.0 , les valeurs `JCEKS.provider` et `JKS.provider` sont ignorées. Le fournisseur Bouncy Castle est utilisé, en conjonction avec la disposition JCE/JCE fournie

par l'environnement d'exécution Java utilisé. Pour plus d'informations, voir «Prise en charge des environnements d'exécution Java nonIBM avec AMS», à la page 631.

Exemples de structures pour les magasins de clés:

système moniteur conversationnel

```
cms.keystore = /dir/keystore_file
cms.certificate = certificate_label
```

PKCS#11

```
pkcs11.library = dir\cryptoki.dll
pkcs11.certificate = certificatelabel
pkcs11.token = tokenlabel
pkcs11.token_pin = tokenpin
pkcs11.secondary_keystore = dir\signers
```

IBM i marque d"erreur d"impression

```
pem.private = /dir/keystore_file_private_key
pem.public = /dir/keystore_file_public_keys
pem.password = password
```

Java JKS

```
jks.keystore = dir/Keystore
jks.certificate = certificate_label
jks.encrypted = no
jks.keystore_pass = password
jks.key_pass = password
jks.provider = IBMJCE
```

Java JCEKS

```
jceks.keystore = dir/Keystore
jceks.certificate = certificate_label
jceks.encrypted = no
jceks.keystore_pass = password
jceks.key_pass = password
jceks.provider = IBMJCE
```

V 9.1.0 Java JCERACFKS

```
jceracfks.keystore = safkeyring://user/keyring
jceracfks.certificate = certificate_label
```

Tableau 98. Récapitulatif des paramètres requis pour chaque type de fichier de configuration

Paramètres	Obligatoire	Type de fichier de configuration			
		V 9.1.0 **Java (JKS, JCEKS et JCERACFKS)**	IBM i **marque d"erreur d"impression**	**PKCS#11**	**système moniteur conversationnel**
keystore	✓	✓			✓
IBM i private	✓		IBM i ✓		

Tableau 98. Récapitulatif des paramètres requis pour chaque type de fichier de configuration (suite)

Paramètres	Obligatoire	Type de fichier de configuration			
		V 9.1.0 Java (JKS, JCEKS et JCERACFKS)	IBM i marque d"erreur d"impression	PKCS#11	système moniteur conversationnel
IBM i public	✓		IBM i ✓		
IBM i password	✓		IBM i ✓		
library	✓			✓	
certificate	✓	✓		✓	✓
token	✓			✓	
token_pin	✓			✓	
secondary_keystore	✓			✓	
encrypted		✓			
keystore_pass	✓	✓			
key_pass		✓			
provider		✓			

Notez que vous pouvez ajouter des commentaires à l'aide du symbole # .

Les paramètres du fichier de configuration sont définis comme suit:

keystore
Configuration CMS et Java uniquement. Chemin d'accès au fichier de clés pour la configuration CMS, JKS et JCEKS.

z/OS V 9.1.0 MQ Adv. VUE URI du fichier de clés RACF pour la configuration de JCERACFKS.

Important :

- Le chemin d'accès au fichier de clés ne doit pas inclure l'extension de fichier.
- z/OS V 9.1.0 MQ Adv. VUE L'URI du fichier de clés RACF doit être au format suivant:

```
safkeyring://user/keyring
```

où :

– *user* est l'ID utilisateur propriétaire du fichier de clés
– *keyring* est le nom du fichier de clés.

IBM i **private**
Configuration PEM uniquement. Nom de fichier d'un fichier contenant une clé privée et un certificat au format PEM.

IBM i **public**
Configuration PEM uniquement. Nom de fichier d'un fichier contenant des certificats publics de confiance au format PEM.

IBM i **password**
Configuration PEM uniquement. Mot de passe utilisé pour déchiffrer une clé privée chiffrée.

library
PKCS#11 uniquement. Nom de chemin de la bibliothèque PKCS#11 .

certificate
Configuration CMS, PKCS#11 et Java uniquement. Label de certificat

token
PKCS#11 uniquement. Libellé de jeton.

token_pin
PKCS#11 uniquement. Code PIN pour déverrouiller le jeton.

secondary_keystore
PKCS#11 uniquement. Nom de chemin du magasin de clés CMS, fourni sans l'extension `.kdb` , qui contient les certificats d'ancrage (certificats racine) requis par les certificats stockés sur le jeton PKCS #11 . Le magasin de clés secondaire peut également contenir des certificats intermédiaires dans la chaîne de confiance, ainsi que des certificats de destinataire définis dans la politique de sécurité de confidentialité. Ce magasin de clés CMS doit être accompagné d'un fichier de dissimulation qui doit se trouver dans le même répertoire que le magasin de clés secondaire.

encrypted
Configuration de Java uniquement. Statut du mot de passe.

keystore_pass
Configuration de Java uniquement. Mot de passe du fichier de clés.

Remarque :

- Pour le magasin de clés CMS, AMS s'appuie sur les fichiers de dissimulation (`.sth`), alors que JKS et JCEKS peuvent nécessiter un mot de passe pour le certificat et la clé privée de l'utilisateur.
- **Important :** Le stockage des mots de passe sous forme de texte en clair représente un risque pour la sécurité.

z/OS V 9.1.0 MQ Adv. VUE

Remarque : Ignoré pour `jceracfks` car l'accès n'est pas contrôlé par un mot de passe.

key_pass
Configuration de Java uniquement. Mot de passe de la clé privée de l'utilisateur.

Important : Le stockage des mots de passe sous forme de texte en clair représente un risque pour la sécurité.

z/OS V 9.1.0 MQ Adv. VUE

Remarque : Ignoré pour `jceracfks` car l'accès n'est pas contrôlé par un mot de passe.

provider
Configuration de Java uniquement. Fournisseur de sécurité Java qui implémente les algorithmes de cryptographie requis par le certificat du magasin de clés.

Important : Les informations stockées dans le magasin de clés sont essentielles pour le flux sécurisé des données envoyées à l'aide de IBM MQ. Les administrateurs de sécurité doivent accorder une attention particulière lorsqu'ils affectent des droits d'accès à ces fichiers.

Exemple de fichier `keystore.conf` :

```
# Native AMS application configuration
cms.keystore = c:\Documents and Settings\Alice\AliceKeystore
cms.certificate = AliceCert

# Java AMS application configurationjceks.keystore = c:/Documents and Settings/Alice/
AliceKeystore
jceks.certificate = AliceCert
jceks.encrypted = no
jceks.keystore_pass = passw0rd
jceks.key_pass = passw0rd
jceks.provider = IBMJCE
```

Tâches associées

«Protection des mots de passe dans Java», à la page 645
Le stockage des mots de passe de clés et de clés privées sous forme de texte en clair représente un risque pour la sécurité. Par conséquent, Advanced Message Security fournit un outil qui peut brouiller ces mots de passe à l'aide de la clé d'un utilisateur, qui est disponible dans le fichier de clés.

Prise en charge des environnements d'exécution Java nonIBM avec AMS

IBM MQ classes for Java et IBM MQ classes for JMS prennent en charge l'opération Advanced Message Security lors de l'exécution avec des environnements d'exécution Java nonIBM .

Advanced Message Security (AMS) implémente Cryptographic Message Syntax (CMS). La syntaxe CMS est utilisée pour signer numériquement, condenser, authentifier ou chiffrer du contenu de message arbitraire.

Depuis la IBM MQ 9.0, le support Advanced Message Security dans IBM MQ classes for Java et IBM MQ classes for JMS utilise les packages open source Bouncy Castle pour prendre en charge CMS. Cela signifie que ces classes peuvent prendre en charge l'opération Advanced Message Security lors de l'exécution avec des environnements d'exécution Java nonIBM .

Avant IBM MQ 9.0, Advanced Message Security n'était pas pris en charge dans les environnements d'exécution Java nonIBM dans les clients Java . La prise en charge de Advanced Message Security dans IBM MQ classes for Java et IBM MQ classes for JMS dépendait de la prise en charge CMS spécifiquement fournie par l'implémentation IBM de JCE (Java Cryptography Extensions). En raison de cette restriction, la fonctionnalité n'était disponible que lors de l'utilisation d'un environnement Java runtime environment (JRE) incluant le fournisseur JCE Java .

Solaris Il est important de noter que la prise en charge sur les plateformes telles que Solaris nécessitait un environnement d'exécution Java hybride, c'est-à-dire l'environnement d'exécution Java standard pour la plateforme avec des éléments supplémentaires fournis par IBM. En particulier, le fournisseur JCE IBM était requis plutôt que le fournisseur JCE fourni par l'environnement d'exécution Java standard pour la plateforme.

Emplacement et numérotation des versions pour les fichiers JAR de Bouncy Castle

Les fichiers JAR Bouncy Castle qui sont requis pour la prise en charge des environnements d'exécution Java nonIBM sont inclus dans le package d'installation IBM MQ classes for Java et IBM MQ classes for JMS .

Les fichiers JAR Bouncy Castle utilisés sont les suivants:

Le fichier JAR du fournisseur, qui est fondamental pour les opérations Bouncy Castle.
Ce fichier JAR est appelé `bcprov-jdk15on.jar`.

Le fichier JAR "PKIX", qui contient la prise en charge des opérations CMS utilisées par Advanced Message Security.
Ce fichier JAR est appelé `bcpkix-jdk15on.jar`.

V 9.1.0.9 **Le fichier JAR "util", qui contient les classes utilisées par les autres fichiers JAR Bouncy Castle.**

Ce fichier JAR est appelé `bcutil-jdk15on.jar`.

Dépendances

Les classes IBM MQ 9.1 et ultérieures ont été testées avec des environnements d'exécution Java IBM et des environnements d'exécution Java Oracle . Ils sont également susceptibles de s'exécuter avec succès dans n'importe quel environnement d'exécution Java J2SE-compliant . Toutefois, vous devez noter les dépendances suivantes:

- Aucune modification n'a été apportée à la configuration de Advanced Message Security .
- Les classes Bouncy Castle sont utilisées uniquement pour les opérations CMS. Toutes les autres opérations liées à la sécurité, par exemple l'accès au magasin de clés, le chiffrement réel des données et le calcul des totaux de contrôle de signature utilisent la fonctionnalité fournie par l'environnement d'exécution Java.

  Important : Pour cette raison, l'environnement d'exécution Java utilisé doit inclure une implémentation de fournisseur JCE.
- Pour utiliser des algorithmes de chiffrement *fort* , vous devrez peut-être installer les fichiers de règles *sans restriction* pour l'implémentation JCE de l'environnement d'exécution Java.

  Pour plus de détails, reportez-vous à la documentation de l'environnement d'exécution Java.
- Si vous avez activé la sécurité Java :
  - Ajoutez `java.security.SecurityPermissioninsertProvider.BC` à l'application pour que les classes Bouncy Castle puissent être utilisées comme fournisseur de sécurité.
  - Accordez `java.security.AllPermission` aux fichiers JAR Bouncy Castle, qui sont les suivants:

    V 9.1.0.9 *mq_install_dir*`/java/lib/bcutil-jdk15on.jar`
    mq_install_dir`/java/lib/bcpkix-jdk15on.jar`
    mq_install_dir`/java/lib/bcprov-jdk15on.jar`

Concepts associés

Ce qui est installé pour IBM MQ classes for JMS
Ce qui est installé pour IBM MQ classes for Java

Multi Interception MCA (Message Channel Agent)

L'interception MCA permet à un gestionnaire de files d'attente s'exécutant sous IBM MQ d'activer de manière sélective les règles à appliquer pour les canaux de connexion serveur.

L'interception MCA permet aux clients qui restent en dehors de AMS d'être toujours connectés à un gestionnaire de files d'attente et de chiffrer et déchiffrer leurs messages.

L'interception MCA est destinée à fournir la fonction AMS lorsque AMS ne peut pas être activé sur le client. Notez que l'utilisation de l'interception MCA et d'un client compatible avec AMSentraîne une double protection des messages qui peut être problématique pour la réception des applications. Pour plus d'informations, voir «Désactivation d'Advanced Message Security sur le client», à la page 635.

Remarque : Les intercepteurs MCA ne sont pas pris en charge pour les canaux AMQP ou MQTT.

Fichier de configuration du magasin de clés

Par défaut, le fichier de configuration du magasin de clés pour l'interception MCA est `keystore.conf` et se trouve dans le répertoire `.mqs` du répertoire HOME de l'utilisateur qui a démarré le gestionnaire de files d'attente ou le programme d'écoute. Le magasin de clés peut également être configuré à l'aide de la variable d'environnement MQS_KEYSTORE_CONF. Pour plus d'informations sur la configuration du magasin de clés AMS , voir «Utilisation de magasins de clés et de certificats», à la page 626.

Pour activer l'interception MCA, vous devez fournir le nom d'un canal que vous souhaitez utiliser dans le fichier de configuration du magasin de clés. Pour l'interception MCA, seul un type de magasin de clés CMS peut être utilisé.

Voir «Exemple d'interception MCA Advanced Message Security», à la page 633 pour un exemple de configuration de l'interception MCA.

Avertissement : Vous devez effectuer l'authentification et le chiffrement des clients sur les canaux sélectionnés, par exemple, en utilisant SSL et SSLPEER ou CHLAUTH TYPE (SSLPEERMAP), pour vous assurer que seuls les clients autorisés peuvent se connecter et utiliser cette fonction.

IBM i

Si votre entreprise utilise IBM iet que vous avez sélectionné une autorité de certification commerciale pour signer votre certificat, le Certificate Manager numérique crée une demande de certificat au format PEM (Privacy-Enhanced Mail). Vous devez transmettre la demande à l'autorité de certification choisie.

Pour ce faire, vous devez utiliser la commande suivante pour sélectionner le certificat approprié pour le canal spécifié dans `channelname`:

```
pem.certificate.channel.channelname
```

Exemple d'interception MCA Advanced Message Security

Exemple de tâche de configuration d'une interception MCA AMS .

Avant de commencer

Avertissement : Vous devez effectuer l'authentification et le chiffrement des clients sur les canaux sélectionnés, par exemple, en utilisant SSL et SSLPEER ou CHLAUTH TYPE (SSLPEERMAP), pour vous assurer que seuls les clients autorisés peuvent se connecter et utiliser cette fonction.

Si votre entreprise utilise IBM iet que vous avez sélectionné une autorité de certification commerciale pour signer votre certificat, le Certificate Manager numérique crée une demande de certificat au format PEM (Privacy-Enhanced Mail). Vous devez transmettre la demande à l'autorité de certification choisie.

Pourquoi et quand exécuter cette tâche

Cette tâche vous guide tout au long du processus de configuration de votre système pour utiliser l'interception MCA, puis de vérification de la configuration.

Remarque : Avant IBM WebSphere MQ 7.5, AMS était un produit complémentaire qui devait être installé séparément et des intercepteurs configurés pour protéger les applications. A partir de IBM WebSphere MQ 7.5 , les intercepteurs sont automatiquement inclus et activés de manière dynamique dans les environnements d'exécution client et serveur MQ . Dans cet exemple d'interception MCA, les intercepteurs sont fournis à l'extrémité serveur du canal et un environnement d'exécution client plus ancien est utilisé (à l'étape 12) pour placer des messages non protégés sur le canal afin qu'ils puissent être considérés comme protégés par les intercepteurs MCA. Si cet exemple avait utilisé un client IBM WebSphere MQ 7.5 ou ultérieur, le message serait protégé deux fois, car l'intercepteur d'exécution du client MQ et l'intercepteur MCA protégeraient le message lorsqu'il arrive dans MQ.

Avertissement : Remplacez `userID` dans le code par votre ID utilisateur.

Procédure

1. Créez la base de données de clés et les certificats à l'aide des commandes suivantes pour créer un script shell.

Modifiez également les paramètres **INSTLOC** et **KEYSTORELOC** ou exécutez les commandes requises. Notez que vous n'avez peut-être pas besoin de créer le certificat pour bob.

```
INSTLOC=/opt/mq90
KEYSTORELOC=/home/testusr/ssl/ams1
mkdir -p $KEYSTORELOC
chmod -R 777 $KEYSTORELOC
chown -R mqm:mqm $KEYSTORELOC
export PATH=$PATH:$INSTLOC/gskit8/bin
echo "PATH = $PATH"
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INSTLOC/gskit8/lib64

gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd -stash
gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd -stash
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd
-label alice_cert -dn "cn=alice,O=IBM,c=IN" -default_cert yes
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd
-label bob_cert  -dn "cn=bob,O=IBM,c=IN"  -default_cert yes
```

2. Partagez les certificats entre les deux bases de données de clés afin que chaque utilisateur puisse identifier l'autre.

   Il est important d'utiliser la méthode décrite dans la tâche 5 de **. Partage de certificats** dans le **Guide de démarrage rapide** (Windows ou UNIX).
3. Créez `keystore.conf` avec la configuration suivante: `Keystore.conf location: /home/userID/ssl/ams1/`

```
cms.keystore = /home/userID/ssl/ams1/alicekey
cms.certificate.channel.SYSTEM.DEF.SVRCONN = alice_cert
```

4. Création et démarrage du gestionnaire de files d'attente AMSQMGR1
5. Définissez un programme d'écoute avec *port* 14567 et *contrôle* QMGR
6. Désactivez les droits d'accès au canal ou définissez les règles relatives aux droits d'accès au canal.

   Pour plus d'informations, voir SET CHLAUTH .
7. Arrêtez le gestionnaire de files d'attente.
8. Définissez le magasin de clés:

```
export MQS_KEYSTORE_CONF=/home/userID/ssl/ams1/keystore.conf
```

9. Démarrez le gestionnaire de files d'attente sur le même interpréteur de commandes.
10. Définissez la stratégie de sécurité et vérifiez:

```
setmqspl -m AMSQMGR1 -s SHA256 -e AES256 -p TESTQ -a "CN=alice,O=IBM,C=IN"
-r "CN=alice,O=IBM,C=IN"
dspmqspl -m AMSQMGR1
```

    Pour plus d'informations, voir setmqspl et dspmqspl .
11. Définissez la configuration de canal:

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/127.0.0.1(14567)'
```

12. Exécutez **amqsputc** à partir d'un client MQ qui n'active pas automatiquement un intercepteur MCA ; par exemple, un client IBM WebSphere MQ 7.1 ou antérieur. Insérez les deux messages suivants:

```
/opt/mqm/samp/bin/amqsputc TESTQ TESTQMGR
```

13. Supprimez la règle de sécurité et vérifiez le résultat:

```
setmqspl -m AMSQMGR1 -p TESTQ -remove
dspmqspl -m AMSQMGR1
```

14. Parcourez la file d'attente à partir de votre installation IBM MQ 9.0 :

```
/opt/mq90/samp/bin/amqsbcg TESTQ AMSQMGR1
```

    La sortie de navigation affiche les messages au format chiffré.

15. Définissez la règle de sécurité et vérifiez le résultat:

```
setmqspl -m AMSQMGR1 -s SHA256 -e AES256 -p TESTQ -a "CN=alice,O=IBM,C=IN"
-r "CN=alice,O=IBM,C=IN"
dspmqspl -m AMSQMGR1
```

16. Exécutez **amqsgetc** à partir de votre installation IBM MQ 9.0 :

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

Tâches associées

«Guide de démarrage rapide pour AMS avec les clients Java», à la page 614
Utilisez ce guide pour configurer rapidement Advanced Message Security afin de garantir la sécurité des messages pour les applications Java qui se connectent à l'aide de liaisons client. Lorsque vous l'aurez terminé, vous aurez créé un magasin de clés pour vérifier les identités utilisateur et défini des règles de signature / chiffrement pour votre gestionnaire de files d'attente.

Référence associée

«Limitations connues de AMS», à la page 584
Un certain nombre d'options IBM MQ ne sont pas prises en charge ou sont soumises à des limitations pour Advanced Message Security.

Désactivation d'Advanced Message Security sur le client

Vous devez désactiver IBM MQ Advanced Message Security (AMS) si vous utilisez un client IBM WebSphere MQ 7.5 ou version ultérieure pour vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit et qu'une erreur 2085 (MQRC_UNKNOWN_OBJECT_NAME) est signalée.

Pourquoi et quand exécuter cette tâche

Depuis la IBM WebSphere MQ 7.5, IBM MQ Advanced Message Security (AMS) est automatiquement activé dans un client IBM MQ . Par conséquent, par défaut, le client tente de vérifier les règles de sécurité pour les objets du gestionnaire de files d'attente. Toutefois, AMS n'est pas activé sur les serveurs des versions antérieures du produit, par exemple IBM WebSphere MQ 7.1, ce qui entraîne le signalement d'une erreur 2085 (MQRC_UNKNOWN_OBJECT_NAME).

Si cette erreur est signalée, lorsque vous tentez de vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit, vous pouvez désactiver AMS comme suit:

- Pour les clients Java, de l'une des façons suivantes :
  - En définissant une variable d'environnement AMQ_DISABLE_CLIENT_AMS.
  - En définissant la propriété système Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS.
  - En utilisant la propriété DisableClientAMS sous la strophe **Security** dans le fichier mqclient.ini.
- Pour les clients C, de l'une des façons suivantes :
  - En définissant une variable d'environnement MQS_DISABLE_ALL_INTERCEPT.
  - En utilisant la propriété DisableClientAMS sous la strophe **Security** dans le fichier mqclient.ini.

Remarque : Dans IBM WebSphere MQ 7.5, vous pouvez également utiliser la variable d'environnement AMQ_DISABLE_CLIENT_AMS. pour les clients C. Depuis la IBM MQ 8.0, vous ne pouvez plus utiliser la variable d'environnement AMQ_DISABLE_CLIENT_AMS pour les clients C. Vous devez utiliser la variable d'environnement MQS_DISABLE_ALL_INTERCEPT à la place.

Procédure

- Pour désactiver AMS sur le client, utilisez l'une des options suivantes:

  Variable d'environnement AMQ_DISABLE_CLIENT_AMS
  Vous devez définir cette variable dans les cas suivants:

- Si vous utilisez un environnement d'exécution Java (JRE) autre que l'environnement d'exécution IBM Java (JRE)
- Si vous utilisez le client IBM WebSphere MQ 7.5 ou version ultérieure IBM MQ classes for JMS ou IBM MQ classes for Java .

Créez la variable d'environnement AMQ_DISABLE_CLIENT_AMS et définissez-la sur TRUE dans l'environnement où l'application s'exécute. Exemple :

```
export AMQ_DISABLE_CLIENT_AMS=TRUE
```

Propriété système Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS
Pour les clients IBM MQ classes for JMS et IBM MQ classes for Java , vous pouvez définir la propriété système Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS sur la valeur TRUE pour l'application Java .

Par exemple, vous pouvez définir la propriété système Java en tant qu'option `-D` lorsque la commande Java est appelée:

```
java -Dcom.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS=TRUE -cp <MQ_INSTALLATION_PATH>/java/lib/
com.ibm.mqjms.jar  my.java.applicationClass
```

Vous pouvez également spécifier la propriété système Java dans un fichier de configuration JMS , `jms.config`, si l'application utilise ce fichier.

variable d'environnement MQS_DISABLE_ALL_INTERCEPT
Vous devez définir cette variable si vous utilisez IBM MQ 8.0 ou une version ultérieure avec des clients natifs et que vous devez désactiver AMS sur le client.

Créez la variable d'environnement MQS_DISABLE_ALL_INTERCEPT et définissez-la sur TRUE dans l'environnement où le client s'exécute. Exemple :

```
export MQS_DISABLE_ALL_INTERCEPT =TRUE
```

Vous pouvez utiliser la variable d'environnement MQS_DISABLE_ALL_INTERCEPT uniquement pour les clients C. Pour les clients Java , vous devez utiliser la variable d'environnement AMQ_DISABLE_CLIENT_AMS à la place.

Propriété `DisableClientAMS` dans le fichier `mqclient.ini`
Vous pouvez utiliser cette option pour les clients IBM MQ classes for JMS et IBM MQ classes for Java et pour les clients C.

Ajoutez le nom de propriété `DisableClientAMS` sous la section **`Security`** du fichier `mqclient.ini` , comme illustré dans l'exemple suivant:

```
Security:
DisableClientAMS=Yes
```

Vous pouvez également activer AMS comme illustré dans l'exemple suivant:

```
Security:
DisableClientAMS=No
```

Que faire ensuite

Pour plus d'informations sur les problèmes liés à l'ouverture de files d'attente protégées AMS , voir Problèmes liés à l'ouverture de files d'attente protégées lors de l'utilisation de AMS avec JMS.

Concepts associés
«Interception MCA (Message Channel Agent)», à la page 632
L'interception MCA permet à un gestionnaire de files d'attente s'exécutant sous IBM MQ d'activer de manière sélective les règles à appliquer pour les canaux de connexion serveur.

Tâches associées
Configuration d'un client à l'aide d'un fichier de configuration

Référence associée

Fichier de configuration IBM MQ classes for JMS

Exigences de certificat pour AMS

Les certificats doivent disposer d'une clé publique RSA pour pouvoir être utilisés avec Advanced Message Security.

Pour plus d'informations sur les différents types de clé publique et pour savoir comment les créer, voir «Certificats numériques et compatibilité CipherSpec dans IBM MQ», à la page 45.

Extensions d'utilisation de clé

Les extensions d'utilisation de clé imposent des restrictions supplémentaires sur la façon dont un certificat peut être utilisé.

Dans Advanced Message Security, l'utilisation des clés des certificats X.509 v3 doit être définie conformément à la spécification RFC 5280.

Pour la qualité de l'intégrité de la protection, si des extensions d'utilisation de clé de certificat sont définies, cet ensemble doit inclure au moins l'une des deux suivantes:

- **nonRepudiation**
- **digitalSignature**

Pour la qualité de la confidentialité de la protection, si des extensions d'utilisation de clé de certificat sont définies, cet ensemble doit inclure:

- **keyEncipherment**

Pour la qualité de la confidentialité de la protection, si des extensions d'utilisation de clé de certificat sont définies, cet ensemble doit inclure:

- **dataEncipherment**

L'utilisation étendue des clés permet d'affiner davantage les extensions d'utilisation des clés. Pour toutes les qualités de protection, si l'utilisation de la clé étendue du certificat est définie, l'ensemble doit inclure:

- **emailProtection**

Concepts associés

«Qualité de protection», à la page 657
Les règles de protection des données Advanced Message Security impliquent une qualité de protection (QOP).

Méthodes de validation de certificat dans AMS

Vous pouvez utiliser Advanced Message Security pour détecter et rejeter les certificats révoqués afin que les messages de vos files d'attente ne soient pas protégés à l'aide de certificats qui ne répondent pas aux normes de sécurité.

AMS vous permet de vérifier la validité d'un certificat à l'aide du protocole OCSP (Online Certificate Status Protocol) ou de la liste de révocation de certificat (CRL).

AMS peut être configuré pour la vérification OCSP et/ou CRL. Si les deux méthodes sont activées, AMS utilise d'abord le protocole OCSP pour le statut de révocation pour des raisons de performances. Si le statut de révocation d'un certificat est indéterminé après la vérification OCSP, AMS utilise la vérification CRL.

Notez que la vérification OCSP et CRL sont activées par défaut.

Concepts associés

«Online Certificate Status Protocol (OCSP) dans AMS», à la page 638
Le protocole OCSP (Online Certificate Status Protocol) détermine si un certificat a été révoqué et, par conséquent, permet de déterminer si le certificat est digne de confiance. OCSP est activé par défaut.

«Listes de révocation de certificat (CRL) dans AMS», à la page 640
Les listes CRL contiennent une liste de certificats qui ont été marqués par l'autorité de certification comme n'étant plus dignes de confiance pour diverses raisons, par exemple, la clé privée a été perdue ou compromise.

Online Certificate Status Protocol (OCSP) dans AMS

Le protocole OCSP (Online Certificate Status Protocol) détermine si un certificat a été révoqué et, par conséquent, permet de déterminer si le certificat est digne de confiance. OCSP est activé par défaut.

OCSP n'est pas pris en charge sur les sytems IBM i .

Activation de la vérification OCSP dans les intercepteurs natifs de Advanced Message Security

La restitution du protocole OCSP (Online Certificate Status Protocol) dans Advanced Message Security est activée par défaut, en fonction des informations contenues dans les certificats utilisés.

Procédure

Ajoutez les options suivantes au fichier de configuration du magasin de clés :

Remarque : Toutes les strophes OCSP sont facultatives et peuvent être spécifiées indépendamment.

Option	Description
`ocsp.enable=off`	Activez la vérification OCSP si le certificat vérifié possède une extension AIA (Authority Info Access) avec une méthode d'accès PKIX_AD_OCSP contenant un URI de l'emplacement du répondeur OCSP. Valeurs possibles: on ou `off`.
`ocsp.url=`*`responder_URL`*	Adresse URL du canal répondeur OCSP. Si cette option est omise, la vérification OCSP non AIA est désactivée.
`ocsp.http.proxy.host=`*`OCSP_proxy`*	Adresse URL du serveur proxy OCSP. Si cette option est omise, un proxy n'est pas utilisé pour les vérifications de certificats en ligne non AIA.
`ocsp.http.proxy.port=`*`port_number`*	Numéro de port du serveur proxy OCSP. Si cette option est omise, le port par défaut 8080 est utilisé.
`ocsp.nonce.generation=on/off`	Génère une valeur nonce lors de l'interrogation d'OCSP. La valeur par défaut est `off`.
`ocsp.nonce.check=on/off`	Vérifie la valeur nonce après la réception d'une réponse d'OCSP. La valeur par défaut est `off`.
`ocsp.nonce.size=8`	Taille de la valeur nonce en octets.
`ocsp.http.get=on/off`	Spécifie HTTP GET comme méthode d'interrogation. Si cette option est définie sur `off`, HTTP POST est utilisé. La valeur par défaut est `off`.
`ocsp.max_response_size=20480`	Taille maximale de la réponse (en octets) du canal répondeur OCSP fourni.

Option	Description
`ocsp.cache_size=100`	Active la mise en cache de la réponse OCSP interne et définit la limite du nombre d'entrées du cache.
`ocsp.timeout=30`	Temps d'attente d'une réponse serveur (en secondes) après laquelle Advanced Message Security expire.
`ocsp.unknown=ACCEPT`	Définit le comportement lorsqu'un serveur OCSP ne peut pas être atteint dans un délai imparti. Valeurs possibles : • ACCEPT Permet le certificat • WARN Permet le certificat et consigne un avertissement • REJECT Empêche l'utilisation du certificat et consigne une erreur

Activation de la restitution OCSP dans Java dans AMS
Pour activer la vérification OCSP pour Java dans Advanced Message Security, modifiez le fichier `java.security` ou le fichier de configuration du magasin de clés.

Pourquoi et quand exécuter cette tâche

Il existe deux façons d'activer la restitution OCSP dans Advanced Message Security:

Utilisation de java.security
Vérifiez si votre certificat contient une extension de certificat AIA (Authority Information Access).

Procédure

1. Si AIA n'est pas configuré ou si vous souhaitez remplacer votre certificat, éditez le fichier `$JAVA_HOME/lib/security/java.security` avec les propriétés suivantes:

   ```
   ocsp.responderURL=http://url.to.responder:port
   ocsp.responderCertSubjectName=CN=Example CA,O=IBM,C=US
   ```

   et activez la vérification OCSP en éditant le fichier `$JAVA_HOME/lib/security/java.security` avec la ligne suivante:

   ```
   ocsp.enable=true
   ```

2. Si AIA est configuré, activez la vérification OCSP en éditant le fichier `$JAVA_HOME/lib/security/java.security` avec la ligne suivante:

   ```
   ocsp.enable=true
   ```

Que faire ensuite

Si vous utilisez Java Security Manager, effectuez également la configuration, ajoutez le droit Java suivant à `lib/security/java.policy`

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
```

Utilisation de keystore.conf

Procédure

Ajoutez l'attribut suivant au fichier de configuration:

```
ocsp.enable=true
```

Important : La définition de cet attribut dans le fichier de configuration remplace les paramètres java.security .

Que faire ensuite

Pour terminer la configuration, ajoutez les droits Java suivants à `lib/security/java.policy`:

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
permission java.security.SecurityPermission "setProperty.ocsp.enable";
```

Listes de révocation de certificat (CRL) dans AMS

Les listes CRL contiennent une liste de certificats qui ont été marqués par l'autorité de certification comme n'étant plus dignes de confiance pour diverses raisons, par exemple, la clé privée a été perdue ou compromise.

Pour valider les certificats, Advanced Message Security construit une chaîne de certificats qui se compose du certificat du signataire et de la chaîne de certificats de l'autorité de certification jusqu'à un point d'ancrage digne de confiance. Un point d'ancrage digne de confiance est un fichier de clés certifiées qui contient un certificat digne de confiance ou un certificat racine digne de confiance utilisé pour vérifier la confiance d'un certificat. AMS vérifie le chemin du certificat à l'aide d'un algorithme de validation PKIX. Lorsque la chaîne est créée et vérifiée, AMS effectue la validation de certificat qui inclut la validation de la date d'émission et d'expiration de chaque certificat de la chaîne par rapport à la date en cours, en vérifiant si l'extension d'utilisation de clé est présente dans le certificat d'entité finale. Si l'extension est ajoutée au certificat, AMS vérifie si **digitalSignature** ou **nonRepudiation** sont également définis. Si ce n'est pas le cas, le `MQRC_SECURITY_ERROR` est signalé et consigné. Ensuite, AMS télécharge les CRL à partir de fichiers ou de LDAP en fonction des valeurs spécifiées dans le fichier de configuration. Seules les listes de révocation de certificat codées au format DER sont prises en charge par AMS. Si aucune configuration liée à la liste de révocation de certificat n'est trouvée dans le fichier de configuration du magasin de clés, AMS n'effectue aucune vérification de validité de la liste de révocation de certificat. Pour chaque certificat de l'autorité de certification, AMS demande à LDAP des listes de révocation de certificat à l'aide des noms distinctifs d'une autorité de certification pour trouver sa liste de révocation de certificat. Les attributs suivants sont inclus dans la requête LDAP:

```
certificateRevocationList,
certificateRevocationList;binary,
authorityRevocationList,
authorityRevocationList;binary
deltaRevocationList
deltaRevocationList;binary,
```

Remarque : `deltaRevocationList` est pris en charge uniquement lorsqu'il est spécifié en tant que points de distribution.

Activation de la prise en charge de la validation de certificat et de la liste de révocation de certificat dans les intercepteurs natifs

Vous devez modifier le fichier de configuration du magasin de clés afin que Advanced Message Security puisse télécharger des CLR à partir du serveur LDAP (Lightweight Directory Access Protocol).

Pourquoi et quand exécuter cette tâche

IBM i L'activation de la prise en charge de la validation de certificat et de la liste de révocation de certificat dans les intercepteurs natifs n'est pas prise en charge pour Advanced Message Security sur IBM i.

Procédure

Ajoutez les options suivantes au fichier de configuration:

Remarque : Toutes les sections CRL sont facultatives et peuvent être spécifiées indépendamment.

Option	Description
`crl.ldap.host=`*`host_name`*	Nom d'hôte du serveur LDAP.
`crl.ldap.port=`*`port_number`*	Numéro de port du serveur LDAP. Vous pouvez spécifier jusqu'à 11 serveurs. Plusieurs hôtes LDAP sont utilisés pour garantir une reprise en ligne transparente en cas d'échec de la connexion LDAP. Tous les serveurs LDAP doivent être des répliques et contenir les mêmes données. Lorsque l'intercepteur AMS Java parvient à se connecter à un serveur LDAP, il ne tente pas de télécharger des listes de révocation de certificat à partir des serveurs restants fournis.
`crl.cdp=off`	Utilisez cette option pour vérifier ou utiliser les extensions CRLDistributionPoints dans les certificats.
`crl.ldap.version=3`	Numéro de version du protocole LDAP. Valeurs possibles: 2 ou 3.
`crl.ldap.user=cn=`*`username`*	Connectez-vous au serveur LDAP. Si cette valeur n'est pas spécifiée, les attributs CRL dans LDAP doivent être lisibles par tous
`crl.ldap.pass=`*`password`*	Mot de passe du serveur LDAP.
`crl.ldap.cache_lifetime=0`	Durée de vie du cache LDAP en secondes. Valeurs possibles: 0 à 86400.
`crl.ldap.cache_size=50`	Taille du cache LDAP. Cette option ne peut être spécifiée que si la valeur `crl.ldap.cache_lifetime` est supérieure à `0`.
`crl.http.proxy.host=some.host.com`	Port du serveur proxy HTTP pour l'extraction de la liste de révocation de certificat CDP.
`crl.http.proxy.port=8080`	Numéro de port du serveur proxy HTTP.
`crl.http.max_response_size=204800`	Taille maximale de la CRL , en octets, qui peut être extraite d'un serveur HTTP accepté par GSKit.
`crl.http.timeout=30`	Délai d'attente d'une réponse du serveur, en secondes, après lequel AMS arrive à expiration.
`crl.http.cache_size=0`	Taille du cache HTTP, en octets.

Option	Description
`crl.unknown=ACCEPT`	Définit le comportement lorsqu'un serveur CRL ne peut pas être atteint dans un délai imparti. Valeurs possibles : • ACCEPT Permet le certificat • WARN Permet le certificat et consigne un avertissement • REJECT Empêche l'utilisation du certificat et consigne une erreur

Activation de la prise en charge de la liste de révocation de certificat dans Java dans AMS
Pour activer la prise en charge de CRL dans Advanced Message Security, vous devez modifier le fichier de configuration du magasin de clés afin de permettre à AMS de télécharger des CRL à partir du serveur LDAP (Lightweight Directory Access Protocol) et de configurer le fichier java.security .

Procédure

1. Ajoutez les options suivantes au fichier de configuration:

En-tête	Description
`crl.ldap.host=`*`host_name`*	Nom d'hôte LDAP.
`crl.ldap.port=`*`port_number`*	Numéro de port du serveur LDAP. Vous pouvez spécifier jusqu'à 11 serveurs. Plusieurs hôtes LDAP sont utilisés pour garantir une reprise en ligne transparente en cas d'échec de la connexion LDAP. Tous les serveurs LDAP doivent être des répliques et contenir les mêmes données. Lorsque l'intercepteur AMS Java parvient à se connecter à un serveur LDAP, il ne tente pas de télécharger des listes de révocation de certificat à partir des serveurs restants fournis. Java n'utilise pas les valeurs `crl.ldap.user` et crl.ldaworldp.pass . Il n'utilise pas d'utilisateur ni de mot de passe lors de la connexion à un serveur LDAP. Par conséquent, les attributs CRL dans LDAP doivent être lisibles par tous.
`crl.cdp=on/off`	Utilisez cette option pour vérifier ou utiliser les extensions CRLDistributionPoints dans les certificats.

2. Modifiez le fichier `JRE/lib/security/java.security` avec les propriétés suivantes:

Nom de la propriété	Description
`com.ibm.security.enableCRLDP`	Cette propriété prend les valeurs suivantes: `true`, `false`. Si la valeur est `true`, lors de la vérification de la révocation de certificat, les listes de révocation de certificat sont localisées à l'aide de l'URL de l'extension des points de distribution de liste de révocation de certificat du certificat. S'il est défini sur `false` ou s'il n'est pas défini, la vérification de la liste de révocation de certificat à l'aide de l'extension des points de distribution de liste de révocation de certificat est désactivée.
`ibm.security.certpath.ldap.cache.lifetime`	Cette propriété peut être utilisée pour définir la durée de vie des entrées dans le cache mémoire de CertStore LDAP sur une valeur en secondes. La valeur 0 désactive le cache ; -1 signifie une durée de vie illimitée. S'il n'est pas défini, la durée de vie par défaut est de 30 secondes.
`com.ibm.security.enableAIAEXT`	Cette propriété prend les valeurs suivantes: `true`, `false`. Si la valeur est `true`, toutes les extensions d'accès aux informations d'autorité trouvées dans les certificats du chemin de certificat en cours de génération sont examinées afin de déterminer si elles contiennent des URI LDAP. Pour chaque URI LDAP trouvé, un objet LDAPCertStore est créé et ajouté à la collection CertStores qui est utilisée pour localiser les autres certificats requis pour générer le chemin de certificat. S'il est défini sur `false` ou s'il n'est pas défini, des objets LDAPCertStore supplémentaires ne sont pas créés.

z/OS *Activation des listes de révocation de certificats (CRL) sous z/OS*

Advanced Message Security prend en charge la vérification CRL (Certificate Revocation List) des certificats numériques utilisés pour protéger les messages de données

Pourquoi et quand exécuter cette tâche

Lorsque cette option est activée, Advanced Message Security valide les certificats de destinataire lorsque les messages sont placés dans une file d'attente protégée contre la confidentialité et valide les certificats d'expéditeur lorsque les messages sont extraits d'une file d'attente protégée (intégrité ou confidentialité). Dans ce cas, la validation comprend la vérification que les certificats pertinents ne sont pas enregistrés dans une LCR pertinente.

Advanced Message Security utilise les services IBM System SSL pour valider les certificats d'expéditeur et de destinataire. La documentation détaillée relative à la validation du certificat SSL du système est disponible dans le manuel z/OS Cryptographic Services System Secure Sockets Layer Programming (SC24-5901).

Pour activer la vérification CRL, vous devez spécifier l'emplacement d'un fichier de configuration CRL via la définition de données CRLFILE dans le JCL de la tâche démarrée pour l'espace adresse AMS. Un exemple de fichier de configuration CRL pouvant être personnalisé est fourni dans *thlqual*.SCSQPROC (CSQ40CRL). Les paramètres autorisés dans ce fichier sont les suivants:

Tableau 99. Variables de configuration CRL Advanced Message Security		
Variable	**Valeur valides**	**Description**
crl.ldap.host[.n]	*hostname -or-hostname: port*	Adresse IP / nom d'hôte de votre serveur LDAP qui héberge les CRL de vos certificats d'émetteur. Si vous n'indiquez pas de numéro de port pour votre serveur LDAP, le numéro de port spécifié par crl.ldap.port est utilisé.
crl.ldap.port	*port*	Numéro de port TCP/IP de votre serveur LDAP.
crl.ldap.user	*ldap_user*	Nom d'utilisateur LDAP à utiliser lors de la connexion au serveur LDAP.
crl.ldap.pass	*mot_de_passe_ldap*	Mot de passe LDAP associé à crl.ldap.user.

Vous pouvez spécifier plusieurs noms d'hôte et ports de serveur LDAP comme suit:

```
crl.ldap.host.1 = hostname -or hostname:port
crl.ldap.host.2 = hostname -or hostname:port
crl.ldap.host.3 = hostname -or hostname:port
```

Vous pouvez spécifier jusqu'à 10 noms d'hôte. Si vous n'indiquez pas de numéro de port pour vos serveurs LDAP, le numéro de port spécifié par crl.ldap.port est utilisé. Chaque serveur LDAP doit utiliser la même combinaison crl.ldap.user/password pour l'accès.

Lorsque la définition de données CRLFILE est spécifiée, la configuration est chargée lors de l'initialisation de l'espace adresse Advanced Message Security et la vérification de la liste de révocation de certificat est activée. Si la définition de données CRLFILE n'est pas spécifiée, ou si le fichier de configuration de la liste de révocation de certificat n'est pas disponible ou n'est pas valide, la vérification de la liste de révocation de certificat est désactivée.

AMS effectue une vérification CRL à l'aide des services de validation de certificat SSL IBM System comme suit:

Tableau 100. Vérifications de la liste de révocation de certificat Advanced Message Security		
Opération	**Qualité de protection**	**Certificat (s) vérifié (s)**
PUT	Confidentialité	Destinataire(s)
GET	Intégrité / Confidentialité	Emetteur

Si une opération de message échoue, une vérification CRL Advanced Message Security effectue les actions suivantes:

Tableau 101. Comportement d'échec de vérification de la liste de révocation de certificat Advanced Message Security	
Opération	**Echec de la vérification CRL**
PUT	Le message n'est pas inséré dans la file d'attente cible. Le code achèvement MQCC_FAILED et le code anomalie MQRC_SECURITY_ERROR sont renvoyés à l'application.

Tableau 101. Comportement d'échec de vérification de la liste de révocation de certificat Advanced Message Security (suite)

Opération	Echec de la vérification CRL
GET	Le message est supprimé de la file d'attente cible et déplacé vers la file d'attente d'erreurs de protection du système. Le code achèvement MQCC_FAILED et le code anomalie MQRC_SECURITY_ERROR sont renvoyés à l'application.

AMS for z/OS utilise les services IBM System SSL pour valider les certificats, ce qui inclut la liste de révocation de certificat et la vérification de la confiance. IBM System SSL fournit la variable d'environnement GSK_CRL_SECURITY_LEVEL pour modérer l'opération de vérification de la liste de révocation de certificat. Exemple :

```
GSK_CRL_SECURITY_LEVEL=MEDIUM
```

Cette variable est documentée dans le manuel z/OS Cryptographic Services System Secure Sockets Layer Programming. Les affectations valides sont les suivantes:

- LOW-La validation du certificat n'échoue pas si le serveur LDAP ne peut pas être contacté.
- MEDIUM-La validation de certificat requiert que le serveur LDAP puisse être contacté, mais ne nécessite pas qu'une liste de révocation de certificat soit définie.
- ELEVE-La validation de certificat requiert que le serveur LDAP puisse être contacté et qu'une liste de révocation de certificat soit définie.

La valeur par défaut de IBM System SSL est MEDIUM. Vous pouvez définir cette variable dans le fichier de configuration spécifié via la définition de données ENVARS dans le JCL de la tâche démarrée pour l'espace adresse AMS. Un exemple de fichier de configuration de variables d'environnement est fourni dans *thlqual*.SCSQPROC (CSQ40ENV).

Remarque : Il incombe aux administrateurs de s'assurer que les services LDAP appropriés sont disponibles et de gérer les entrées de liste de révocation de certificat pour les autorités de certification concernées.

Protection des mots de passe dans Java

Le stockage des mots de passe de clés et de clés privées sous forme de texte en clair représente un risque pour la sécurité. Par conséquent, Advanced Message Security fournit un outil qui peut brouiller ces mots de passe à l'aide de la clé d'un utilisateur, qui est disponible dans le fichier de clés.

Avant de commencer

Le propriétaire du fichier `keystore.conf` doit s'assurer que seul le propriétaire du fichier est autorisé à lire le fichier. La protection par mot de passe décrite dans ce chapitre n'est qu'une mesure de protection supplémentaire.

Procédure

1. Editez les fichiers `keystore.conf` pour inclure le chemin d'accès au magasin de clés et le libellé des utilisateurs.

   ```
   jceks.keystore = c:/Documents and Setting/Alice/AliceKeystore
   jceks.certificate = AliceCert
   jceks.provider = IBMJCE
   ```

2. Pour exécuter l'outil, exécutez:

```
java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
private_key_password
```

Une sortie avec des mots de passe chiffrés est générée et peut être copiée dans le fichier `keystore.conf`.

Pour copier automatiquement la sortie dans le fichier `keystore.conf`, exécutez:

```
java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
private_key_password >> ~/path_to_keystore/keystore.conf
```

Remarque :

Pour obtenir la liste des emplacements par défaut de `keystore.conf` sur différentes plateformes, voir «Utilisation de magasins de clés et de certificats», à la page 626.

Exemple

Voici un exemple de sortie de ce type:

```
#Fri Jul 30 15:20:29 CEST 2010
jceks.key_pass=MMXh997n5ZOr8uRlJmc5qity9MN2CggGBMKCDxdbn1AyPklvdgTsOLG6X3C1YT7oDzwaqZFlOR4t\r\nm
Zsc7JGAx8nqqxLnAucdGn0NWo6xnjZB1n501YGol2k/
PhaQHhFXKMAU9dKg0f8djOtCAOlX4ETe\r\nfYl9LBUt2wk87uM7dSs\=
jceks.keystore_pass=OIdeayBnSCfLG4cFuxEVrk6SYyAsdSPpDqgPf16s9s1M04cqZjNbhgjoA2EXonudHZHH+4s2drvQ
\r\nCUvQgu9GuaBMJK2F2OjtHJJ1Y4BVeLW2c2okgawo/
W2J1AdUYKkJ0raYTkDouLaTYTQeulyG0xIl\r\niD2si1xUCxhYvvyhbbY\=
jceks.encrypted=yes
```

z/OS Utilisation de certificats sous z/OS

Pourquoi et quand exécuter cette tâche

Advanced Message Security implémente trois niveaux de protection: l'intégrité, la confidentialité et la confidentialité.

Avec une règle d'intégrité, les messages sont signés à l'aide de la clé privée de l'émetteur (l'application effectuant le MQPUT). L'intégrité permet de détecter la modification des messages, mais le texte du message lui-même n'est pas chiffré.

Avec une règle de confidentialité, le message est chiffré lorsqu'il est inséré dans la file d'attente. Le message est chiffré à l'aide d'une clé symétrique et d'un algorithme spécifié dans la règle Advanced Message Security appropriée. La clé symétrique elle-même est chiffrée avec la clé publique de chaque destinataire (l'application effectuant la commande MQGET). Les clés publiques sont associées à des certificats stockés dans des fichiers de clés.

Avec une politique de confidentialité, les messages sont à la fois signés et chiffrés.

Lorsqu'un message protégé par la confidentialité est retiré de la file d'attente par une application destinataire qui effectue une opération MQGET, le message doit être déchiffré. Comme il a été chiffré à l'aide de la clé publique du destinataire, il doit être déchiffré à l'aide de la clé privée du destinataire trouvée dans un fichier de clés.

z/OS *Utilisation des fichiers de clés SAF*

Advanced Message Security (AMS) utilise les services de fichier de clés SAF z/OS pour définir et gérer les certificats nécessaires à la signature et au chiffrement. Les produits de sécurité qui sont fonctionnellement équivalents à RACF peuvent être utilisés à la place de RACF s'ils fournissent le même niveau de prise en charge.

L'utilisation efficace des fichiers de clés peut réduire l'administration nécessaire à la gestion des certificats.

Une fois qu'un certificat est généré (ou importé), il doit être connecté à un fichier de clés pour être accessible. Le même certificat peut être connecté à plusieurs clés.

Advanced Message Security utilise deux ensembles de fichiers de clés. Un ensemble est constitué de fichiers de clés appartenant aux ID utilisateur individuels qui sont à l'origine ou qui reçoivent des messages. Chaque fichier de clés contient la clé privée associée au certificat de l'ID utilisateur propriétaire. La clé privée de chaque certificat est utilisée pour signer les messages pour les files d'attente protégées par l'intégrité ou la confidentialité. Il est également utilisé pour déchiffrer les messages des files d'attente protégées par la confidentialité ou par la protection de la confidentialité lors de la réception de messages.

L'autre ensemble est un fichier de clés unique appartenant à l'utilisateur de l'espace adresse AMS . Il contient la chaîne de signature des certificats de l'autorité de certification nécessaires pour valider les certificats de l'émetteur et des destinataires du message.

Lorsque la protection de la confidentialité ou de la confidentialité est utilisée, le fichier de clés appartenant à l'utilisateur de l'espace adresse AMS contient également les certificats des destinataires du message. Les clés publiques de ces certificats sont utilisées pour chiffrer la clé symétrique qui a été utilisée pour chiffrer les données de message lors de l'insertion du message dans la file d'attente protégée. Lorsque ces messages sont extraits, la clé privée des destinataires concernés est utilisée pour déchiffrer la clé symétrique qui est ensuite utilisée pour déchiffrer les données du message.

Advanced Message Security utilise le nom de fichier de clés **drq.ams.keyring** lors de la recherche de certificats et de clés privées. C'est le cas à la fois pour l'utilisateur et pour les fichiers de clés de l'espace adresse AMS .

Pour une illustration et une explication plus détaillée des certificats et des fichiers de clés, ainsi que de leur rôle dans la protection des données, voir Récapitulatif des opérations liées aux certificats.

La clé privée utilisée pour la signature et le déchiffrement peut avoir n'importe quel libellé mais doit être connectée en tant que certificat par défaut.

Les certificats numériques et les fichiers de clés sont gérés dans RACF principalement à l'aide de la commande RACDCERT.

Pour plus d'informations sur les certificats, les libellés et la commande RACDCERT, voir *z/OS: Security Server RACF Command Language Reference* et *z/OS: Security Server RACF Security Administrator's Guide*.

z/OS *Autorisation d'accès à la commande RACDCERT*

L'autorisation d'utiliser la commande RACDCERT est une tâche de post-installation qui aurait dû être effectuée par votre programmeur système z/OS . Cette tâche implique l'octroi de droits appropriés à l'administrateur de la sécurité Advanced Message Security .

En résumé, ces commandes sont nécessaires pour permettre l'accès à la commande RACDCERT RACF :

```
RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID( admin ) ACCESS(CONTROL)
SETROPTS RACLIST(FACILITY) REFRESH
```

Dans cet exemple, *admin* indique l'ID utilisateur de votre administrateur de sécurité ou de tout utilisateur que vous souhaitez utiliser la commande RACDCERT.

z/OS *Création des certificats et des fichiers de clés*

Cette section décrit les étapes requises pour créer les certificats et les fichiers de clés nécessaires aux utilisateurs z/OS de Advanced Message Security (AMS), à l'aide d'une autorité de certification (CA) RACF .

Résolution des problèmes liés aux certificats lors de l'utilisation de Advanced Message Security sous z/OS

Si vous rencontrez des problèmes avec des certificats et des entrées manquantes dans les magasins de clés, vous pouvez activer une trace GSKIT.

Dans le fichier référencé par la définition de données ENVARS dans la procédure de tâche démarrée AMS , ajoutez:

```
GSK_TRACE_FILE=/u/... /gsktrace
GSK_TRACE=0xff
```

Pour plus d'informations, voir Variables d'environnement .

Pour chaque accès au magasin de clés, les données sont écrites dans le fichier de trace spécifié dans GSK_TRACE_FILE.

Pour formater le fichier de trace, utilisez la commande suivante:

```
gsktrace inputtrace file > output_file
```

Scénario

Un scénario d'application d'envoi et d'application de réception est utilisé pour expliquer les étapes requises.

Dans les exemples qui suivent, user1 est l'émetteur d'un message et user2 est le destinataire. L'ID utilisateur de l'espace adresse Advanced Message Security est WMQAMSD.

Toutes les commandes des exemples présentés ici sont émises à partir de l'option ISPF 6 par l'ID administrateur admin.

z/OS *Définition d'un certificat d'autorité de certification local*

Si vous utilisez RACF comme autorité de certification, vous devez créer un certificat d'autorité de certification, si ce n'est pas déjà fait. La commande affichée ici crée un certificat d'autorité de certification (ou de signataire). Cet exemple crée un certificat appelé AMSCA à utiliser lors de la création de certificats ultérieurs qui reflètent l'identité des utilisateurs et des applications Advanced Message Security .

Cette commande peut être modifiée, en particulier SUBJECTSDN, pour refléter la structure de dénomination et les conventions utilisées lors de votre installation:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('AMSCA') O('ibm') C('us'))
KEYUSAGE(CERTSIGN) WITHLABEL('AMSCA')
```

Remarque : Les certificats signés avec ce certificat d'autorité de certification locale affichent un émetteur de CN=AMSCA, O=ibm, C=us lorsqu'ils sont répertoriés avec la commande RACDCERT LIST.

z/OS *Création d'un certificat numérique avec une clé privée*

Un certificat numérique avec une clé privée doit être généré pour chaque utilisateur Advanced Message Security . Dans l'exemple illustré ici, les commandes RACDCERT sont utilisées pour générer des certificats pour user1 et user2, qui sont signés avec le certificat de l'autorité de certification locale identifié par le libellé AMSCA.

```
RACDCERT ID(user1) GENCERT SUBJECTSDN(CN('user1') O('ibm') C('us'))
WITHLABEL('user1') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(user2) GENCERT SUBJECTSDN(CN('user2') O('ibm') C('us'))
WITHLABEL('user2') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

```
RACDCERT ID(user1) ALTER (LABEL('user1')) TRUST
RACDCERT ID(user2) ALTER (LABEL('user2')) TRUST
```

La commande RACDCERT ALTER est requise pour ajouter l'attribut TRUST au certificat. Lorsqu'un certificat est créé pour la première fois à l'aide de cette procédure, il possède une plage de dates valide différente de celle du certificat signataire. Par conséquent, RACF le marque comme NOTRUST, ce qui signifie que le certificat ne doit pas être utilisé. Utilisez la commande RACDCERT ALTER pour définir l'attribut TRUST.

Les attributs KEYUSAGE HANDSHAKE, DATAENCRYPT et DOCSIGN doivent être spécifiés pour les certificats utilisés par Advanced Message Security.

Tableau 102. Valeurs et indicateurs RACDCERT KEYUSAGE

Valeur KEYUSAGE	**Ensemble d'indicateurs**
HANDSHAKE	digitalSignature et keyEncipherment
CHIFFREDONNEES	dataEncipherment
SIGDOCN	nonRepudiation
SIGCERT	keyCertSign et cRLSign

z/OS *Création des fichiers de clés RACF*

Les commandes présentées ici créent un fichier de clés pour les ID utilisateur définis par RACF user1, user2et l'utilisateur WMQAMSD de la tâche d'espace adresse Advanced Message Security . Le nom du fichier de clés est fixé par Advanced Message Security et doit être codé comme indiqué, sans guillemets. Le nom est sensible à la casse.

```
RACDCERT ID(user1) ADDRING(drq.ams.keyring)
RACDCERT ID(user2) ADDRING(drq.ams.keyring)
RACDCERT ID(WMQAMSD) ADDRING(drq.ams.keyring)
```

z/OS *Connexion des certificats aux fichiers de clés*

Connectez les certificats de l'utilisateur et de l'autorité de certification aux fichiers de clés:

```
RACDCERT ID(WMQAMSD) CONNECT(CERTAUTH LABEL('AMSCA')
RING(drq.ams.keyring))
RACDCERT ID(user1) CONNECT(ID(user1) LABEL('user1')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(user2) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(WMQAMSD) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) USAGE(SITE))
```

Le certificat contenant la clé privée utilisée pour le déchiffrement doit être connecté au fichier de clés de l'utilisateur en tant que certificat par défaut.

L'attribut RACDCERT USAGE (SITE) empêche l'accès à la clé privée dans le fichier de clés, tandis que l'attribut RACDCERT USAGE (PERSONAL) permet l'utilisation de la clé privée, si elle existe. Le certificat de User2doit être connecté au fichier de clés de l'espace adresse Advanced Message Security car sa clé publique est nécessaire pour chiffrer les messages lorsqu'ils sont placés dans la file d'attente. USAGE (SITE) limite l'exposition de la clé privée de user2.

Le certificat CERTAUTH avec le libellé AMSCA doit être connecté au fichier de clés de l'espace adresse Advanced Message Security car il a été utilisé pour signer le certificat de user1, qui est l'émetteur du message. Il est utilisé pour valider le certificat signataire de user1.

z/OS *Vérification du fichier de clés*

Le fichier de clés doit apparaître comme indiqué ici, une fois que toutes les commandes ont été entrées:

```
RACDCERT ID(user1) LISTRING(drq.ams.keyring)
Digital ring information for user USER1:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
user1                            ID(USER1)    PERSONAL YES

RACDCERT ID(user2) LISTRING(drq.ams.keyring)
Digital ring information for user USER2:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
user2                            ID(USER2)    PERSONAL YES

RACDCERT ID(WMQAMSD) LISTRING(drq.ams.keyring)
Digital ring information for user WMQAMSD:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
AMSCA                            CERTAUTH     CERTAUTH NO
user2                            ID(USER2)    SITE     NO
```

La liste des certificats individuels indique également l'association d'anneau.

```
RACDCERT ID(user2) LIST(label('user2'))
Digital certificate information for user USER2:

***
Label: user2
Certificate ID: 2QfH8Pny9/LzpKKFmfFA
Status: TRUST
Start Date: 2010/05/03 22:59:53
End Date:  2011/05/04 22:59:52
Serial Number:>15<:
Issuer's Name:>OU=AMSCA.O=ibm.C=us<:
Subject's Name:>CN=user2.O=ibm.C=us<:
Key Usage: HANDSHAKE, DATAENCRYPT, DOCSIGN
Private Key Type: Non-ICSF
Private Key Size: 1024
Ring Associations:
Ring Owner: USER2
Ring:>drq.ams.keyring<:
Ring Owner: WMQAMSD
Ring:>drq.ams.keyring<:
```

Pour améliorer les performances, le contenu du fichier de clés drq.ams.keyring associé à l'espace adresse AMS est mis en cache pendant toute la durée de vie de l'espace adresse. Les modifications apportées à ce fichier de clés ne sont pas prises en compte automatiquement. L'administrateur peut actualiser le cache de l'une des manières suivantes:

- Arrêt et redémarrage du gestionnaire de files d'attente.
- A l'aide de la commande z/OS MODIFY:

```
F qmgrAMSM,REFRESH KEYRING
```

Tâches associées

fonctionnementAdvanced Message Security

z/OS *Récapitulatif des opérations liées aux certificats*

La Figure 35, à la page 651 illustre les relations entre les applications d'envoi et de réception et les certificats appropriés. Le scénario illustré implique la mise en file d'attente à distance entre deux

gestionnaires de files d'attente z/OS à l'aide d'une règle de protection des données de confidentialité. Dans Figure 35, à la page 651, "AMS" indique " Advanced Message Security".

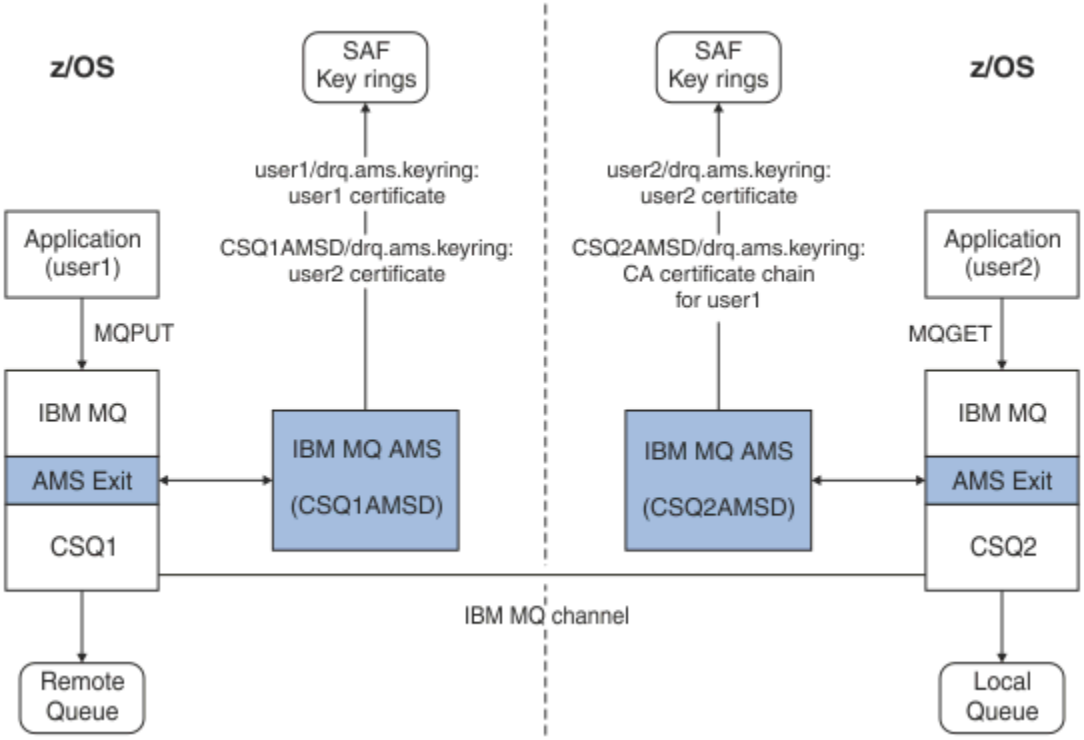


Figure 35. Relations d'application et de certificat

Dans ce diagramme, une application s'exécutant en tant que'user1'place un message dans une file d'attente éloignée gérée par le gestionnaire de files d'attente CSQ1, destinée à être extraite par une application s'exécutant en tant que'user2'à partir d'une file d'attente locale gérée par le gestionnaire de files d'attente CSQ2. Le diagramme suppose une politique de confidentialité Advanced Message Security , ce qui signifie que le message est à la fois signé et chiffré.

Advanced Message Security intercepte le message lorsqu'une insertion se produit et utilise le certificat de user2(stocké dans le fichier de clés de l'utilisateur de l'espace adresse AMS) pour chiffrer une clé symétrique utilisée pour chiffrer les données du message.

Notez que le certificat de user2est connecté au fichier de clés de l'utilisateur de l'espace adresse AMS avec l'option USAGE (SITE). Cela signifie que l'utilisateur de l'espace adresse AMS peut accéder au certificat et à la clé publique, mais pas à la clé privée.

A la fin de la réception, Advanced Message Security intercepte l'obtention émise par user2et utilise le certificat de user2pour déchiffrer la clé symétrique afin qu'elle puisse déchiffrer les données du message. Il valide ensuite la signature de l'utilisateur user1à l'aide de la chaîne de certificats de l'autorité de certification du certificat de l'utilisateur user1stocké dans le fichier de clés de l'utilisateur de l'espace adresse AMS.

Dans ce scénario, mais avec une politique de protection des données d'intégrité, les certificats pour user2 ne sont pas requis.

Pour utiliser Advanced Message Security pour mettre en file d'attente des messages dans des files d'attente protégées par IBM MQayant une politique de protection des messages de confidentialité ou d'intégrité, Advanced Message Security doit avoir accès aux éléments de données suivants:

- Certificat et clé privée X.509 V2 ou V3 pour l'utilisateur qui met le message en file d'attente.
- Chaîne de certificats utilisée pour signer les certificats numériques de tous les signataires de message.

- Si la politique de protection des données est la confidentialité, le certificat X.509 V2 ou V3 des destinataires prévus. Les destinataires prévus sont répertoriés dans la règle Advanced Message Security associée à la file d'attente.

Pour les processus et les applications qui s'exécutent sur z/OS, Advanced Message Security doit avoir des certificats à deux endroits:

- Dans un fichier de clés géré par SAF associé à l'identité RACF de l'application émettrice (l'application qui met en file d'attente le message protégé) ou de l'application réceptrice (si la confidentialité est utilisée).

  Le certificat qui est localisé par Advanced Message Security est le certificat par défaut et doit inclure la clé privée. Advanced Message Security suppose l'identité de l'utilisateur z/OS de l'application émettrice. C'est-à-dire qu'il agit comme un substitut, de sorte qu'il peut accéder à la clé privée de l'utilisateur.
- Dans un fichier de clés géré par SAF associé à l'utilisateur de l'espace adresse AMS.

  Lors de l'envoi de messages protégés par la confidentialité, ce fichier de clés contient les certificats de clé publique des destinataires du message. Lors de la réception de messages, il contient la chaîne de certificats de l'autorité de certification nécessaire à la validation de la signature de l'expéditeur du message.

Les exemples précédents ont utilisé RACF comme autorité de certification locale. Toutefois, vous pouvez utiliser un autre fournisseur PKI (autorité de certification) lors de votre installation. Si vous prévoyez d'utiliser un autre produit PKI, n'oubliez pas que la clé privée et le certificat doivent être importés dans un fichier de clés associé aux ID utilisateur z/OS RACF qui sont à l'origine des messages IBM MQ protégés par Advanced Message Security.

Vous pouvez utiliser la commande RACF RACDCERT comme mécanisme pour générer des demandes de certificat, qui peuvent être exportées et envoyées au fournisseur PKI de votre choix pour être émises.

Voici un récapitulatif des étapes liées aux certificats:

1. Demandez la création d'un certificat d'autorité de certification, dans lequel RACF est l'autorité de certification locale. Omettez cette étape si vous utilisez un autre fournisseur PKI.
2. Générez des certificats d'utilisateur signés par l'autorité de certification.
3. Créez les fichiers de clés pour les utilisateurs et l'ID d'espace adresse AMS Advanced Message Security .
4. Connectez le certificat utilisateur au fichier de clés de l'utilisateur avec l'attribut par défaut.
5. Connectez les certificats des destinataires au fichier de clés de l'utilisateur de l'espace adresse AMS Advanced Message Security à l'aide de l'attribut d'utilisation (site) (cette étape est nécessaire uniquement pour les certificats utilisateur qui seront finalement les destinataires des messages protégés par la confidentialité).
6. Connectez les chaînes de certificats de l'autorité de certification pour les émetteurs de messages au fichier de clés de l'utilisateur de l'espace adresse Advanced Message Security AMS. (Cette étape est nécessaire uniquement pour les tâches AMS qui vérifient les signatures d'expéditeur.)

z/OS *Configuration d'une infrastructure PKI nonz/OS résidente*

Advanced Message Security for z/OSutilise des certificats numériques X.509 V3 dans le traitement de la protection des messages placés ou reçus des files d'attente IBM MQ . Advanced Message Security lui-même ne crée ni ne gère le cycle de vie de ces certificats ; cette fonction est fournie par une infrastructure à clé publique (PKI). Les exemples de cette publication qui illustrent l'utilisation de certificats utilisent z/OS Security Server RACF pour répondre aux demandes de certificat.

Qu'une infrastructure PKI z/OS ou nonz/OS soit utilisée ou non, AMS for z/OS utilise uniquement des fichiers de clés gérés par RACF ou son équivalent. Ces fichiers de clés sont basés sur SAF (Security Authorization Facility) et sont le référentiel utilisé par AMS for z/OS pour extraire les certificats des émetteurs et des destinataires des messages placés dans ou reçus des files d'attente IBM MQ .

Pour les messages provenant de z/OS, qui sont protégés par des règles d'intégrité ou de chiffrement, le certificat et la clé privée de l'ID utilisateur d'origine doivent être stockés dans un fichier de clés géré par SAF associé à l'ID utilisateur z/OS de l'émetteur du message.

RACF inclut la possibilité d'importer des certificats et des clés privées dans des fichiers de clés gérés par RACF. Pour plus de détails et des exemples de chargement de certificats dans des fichiers de clés gérés RACF , voir les publications z/OS Security Server RACF .

Si votre installation utilise l'un des produits PKI pris en charge, reportez-vous aux publications qui accompagnent le produit pour savoir comment le déployer.

Administration des règles de sécurité Advanced Message Security

Advanced Message Security utilise des règles de sécurité pour spécifier les algorithmes de chiffrement cryptographique et de signature pour le chiffrement et l'authentification des messages qui transitent par les files d'attente.

Présentation des stratégies de sécurité pour AMS

Les règles de sécurité Advanced Message Security sont des objets conceptuels qui décrivent la façon dont un message est chiffré et signé de manière cryptographique.

Pour plus de détails sur les attributs de stratégie de sécurité, voir les sous-rubriques suivantes:

Concepts associés

«Qualité de protection», à la page 657
Les règles de protection des données Advanced Message Security impliquent une qualité de protection (QOP).

«Attributs de stratégie de sécurité dans AMS», à la page 656
Vous pouvez utiliser Advanced Message Security pour sélectionner un algorithme ou une méthode spécifique afin de protéger les données.

Noms de règle dans AMS

Le nom de règle est un nom unique qui identifie une règle Advanced Message Security spécifique et la file d'attente à laquelle elle s'applique.

Le nom de la règle doit être identique au nom de la file d'attente à laquelle elle s'applique. Il existe un mappage un à un entre un Advanced Message Security (AMS) et une file d'attente.

En créant une règle portant le même nom qu'une file d'attente, vous activez la règle pour cette file d'attente. Les files d'attente sans noms de règle correspondants ne sont pas protégées par AMS.

La portée de la règle est pertinente pour le gestionnaire de files d'attente local et ses files d'attente. Les gestionnaires de files d'attente éloignées doivent disposer de leurs propres règles définies en local pour les files d'attente qu'ils gèrent.

Algorithme de signature dans AMS

L'algorithme de signature indique l'algorithme qui doit être utilisé lors de la signature des messages de données.

Les valeurs valides sont les suivantes :

- MD5
- SHA-1
- SHA-2 Famille :
  - SHA256
  - SHA384 (longueur de clé minimale acceptable-768 bits)
  - SHA512 (longueur de clé minimale acceptable-768 bits)

Une règle qui ne spécifie pas d'algorithme de signature, ou qui spécifie un algorithme de NONE, implique que les messages placés dans la file d'attente associée à la règle ne sont pas signés.

Remarque : La qualité de protection utilisée pour les fonctions d'insertion et d'obtention de message doit correspondre. S'il existe une non-concordance de la qualité de protection de la règle entre la file d'attente et le message dans la file d'attente, le message n'est pas accepté et est envoyé à la file d'attente de traitement des erreurs. Cette règle s'applique aux files d'attente locales et éloignées.

Algorithme de chiffrement dans AMS

L'algorithme de chiffrement indique l'algorithme qui doit être utilisé lors du chiffrement des messages de données placés dans la file d'attente associée à la règle.

Les valeurs valides sont les suivantes :

- RC2
- DES
- 3DES
- AES128
- AES256

Une règle qui ne spécifie pas d'algorithme de chiffrement ou qui spécifie un algorithme de NONE implique que les messages placés dans la file d'attente associée à la règle ne sont pas chiffrés.

Notez qu'une règle qui spécifie un algorithme de chiffrement autre que NONE doit également spécifier au moins un nom distinctif de destinataire et un algorithme de signature car les messages chiffrés Advanced Message Security sont également signés.

Important : La qualité de protection utilisée pour les fonctions d'insertion et d'obtention de message doit correspondre. S'il existe une non-concordance de la qualité de protection de la règle entre la file d'attente et le message dans la file d'attente, le message n'est pas accepté et est envoyé à la file d'attente de traitement des erreurs. Cette règle s'applique aux files d'attente locales et éloignées.

Tolérance dans AMS

L'attribut toleration indique si Advanced Message Security peut accepter les messages pour lesquels aucune règle de sécurité n'est spécifiée.

Lors de l'extraction d'un message d'une file d'attente avec une règle de chiffrement des messages, si le message n'est pas chiffré, il est renvoyé à l'application appelante. Les valeurs valides sont les suivantes :

0
Non (**par défaut**).

1
Oui.

Une règle qui ne spécifie pas de valeur de tolérance ou qui indique 0 implique que les messages placés dans la file d'attente associée à la règle doivent correspondre aux règles de la règle.

La tolérance est facultative et existe pour faciliter le déploiement de la configuration, où les règles ont été appliquées aux files d'attente, mais ces dernières contiennent déjà des messages pour lesquels aucune règle de sécurité n'est spécifiée.

Noms distinctifs d'expéditeur dans AMS

Les noms distinctifs d'expéditeur identifient les utilisateurs autorisés à placer des messages dans une file d'attente. Un expéditeur utilise son certificat pour signer un message, avant de le placer dans une file d'attente.

Advanced Message Security (AMS) ne vérifie pas si un message a été placé dans une file d'attente protégée par des données par un utilisateur valide jusqu'à ce que le message soit extrait. A ce stade, si la règle stipule un ou plusieurs expéditeurs valides et que l'utilisateur qui a placé le message dans la file d'attente ne figure pas dans la liste des expéditeurs valides, AMS renvoie une erreur à l'application de réception et place le message dans la file d'attente d'erreurs AMS.

Une règle peut avoir zéro ou plusieurs noms distinctifs d'expéditeurs spécifiés. Si aucun nom distinctif d'expéditeur n'est spécifié pour la règle, tout expéditeur peut placer des messages protégés par des données dans la file d'attente à condition que le certificat de l'expéditeur soit digne de confiance. Le certificat d'un expéditeur est digne de confiance en ajoutant le certificat public à un magasin de clés disponible pour l'application de réception.

Les noms distinctifs des expéditeurs se présentent sous la forme suivante :

```
CN=Common Name,O=Organization,C=Country
```

Important :

- Tous les noms distinctifs doivent être en majuscules. Tous les identificateurs de nom de composant du nom distinctif doivent être indiqués dans l'ordre indiqué dans le tableau suivant:

Nom de composant	Valeur
CN	Nom usuel de l'objet de ce nom distinctif, tel qu'un nom complet ou la finalité prévue d'un périphérique.
OU	Unité au sein de l'organisation à laquelle l'objet du nom distinctif est affilié, telle qu'une division d'entreprise ou un nom de produit.
O	Organisation à laquelle l'objet du nom distinctif est affilié, telle qu'une société.
L	La localité (ville ou municipalité) où se trouve l'objet du nom distinctif.
ST	Nom de l'état ou de la province où se trouve l'objet du nom distinctif.
C	Pays dans lequel se trouve l'objet du nom distinctif (DN).

- Si un ou plusieurs noms distinctifs d'expéditeur sont spécifiés pour la règle, seuls ces utilisateurs peuvent placer des messages dans la file d'attente associée à la règle.
- Les noms distinctifs d'expéditeur, lorsqu'ils sont spécifiés, doivent correspondre exactement aux noms distinctifs contenus dans le certificat numérique associé à l'utilisateur plaçant le message.
- AMS prend en charge les noms distinctifs dont les valeurs proviennent uniquement du jeu de caractères Latin-1 . Pour créer des noms distinctifs avec des caractères de l'ensemble, vous devez d'abord créer un certificat avec un nom distinctif créé en UTF-8 à l'aide de UNIX avec le codage UTF-8 activé ou avec l'interface graphique **strmqikm** . Vous devez ensuite créer une règle à partir d'une plateforme UNIX avec le codage UTF-8 activé ou utiliser le plug-in AMS dans IBM MQ.
- La méthode utilisée par AMS, pour convertir le nom de l'expéditeur du format x.509 au format DN, utilise toujours ST = pour la valeur Etat ou Province.
- Les caractères spéciaux suivants nécessitent des caractères d'échappement:

```
, (comma)
+ (plus)
" (double quote)
\ (backslash)
< (less than)
> (greater than)
; (semicolon)
```

- Si le nom distinctif contient des blancs imbriqués, vous devez placer le nom distinctif entre guillemets.

Concepts associés

«Noms distinctifs des destinataires dans AMS», à la page 656

Les noms distinctifs des destinataires identifient les utilisateurs qui sont autorisés à extraire des messages d'une file d'attente.

Noms distinctifs des destinataires dans AMS

Les noms distinctifs des destinataires identifient les utilisateurs qui sont autorisés à extraire des messages d'une file d'attente.

Une règle peut avoir zéro ou plusieurs noms distinctifs de destinataires spécifiés. Les noms distinctifs des destinataires se présentent sous la forme suivante:

```
CN=Common Name,O=Organization,C=Country
```

Important :

- Tous les noms distinctifs doivent être en majuscules. Tous les identificateurs de nom de composant du nom distinctif doivent être indiqués dans l'ordre indiqué dans le tableau suivant:

Nom de composant	Valeur
CN	Nom usuel de l'objet de ce nom distinctif, tel qu'un nom complet ou la finalité prévue d'un périphérique.
OU	Unité au sein de l'organisation à laquelle l'objet du nom distinctif est affilié, telle qu'une division d'entreprise ou un nom de produit.
O	Organisation à laquelle l'objet du nom distinctif est affilié, telle qu'une société.
L	La localité (ville ou municipalité) où se trouve l'objet du nom distinctif.
ST	Nom de l'état ou de la province où se trouve l'objet du nom distinctif.
C	Pays dans lequel se trouve l'objet du nom distinctif (DN).

- Si aucun nom distinctif de destinataire n'est spécifié pour la règle, tous les utilisateurs peuvent récupérer des messages de la file d'attente associée aux règles.
- Si un ou plusieurs noms distinctifs de destinataire est (sont) spécifié(s) pour la règle, seuls ces utilisateurs peuvent récupérer des messages de la file d'attente associée aux règles.
- Les noms distinctifs de destinataire, lorsqu'ils sont spécifiés, doivent correspondre exactement au nom distinctif contenu dans le certificat numérique associé à l'utilisateur récupérant le message.
- Advanced Message Security prend en charge les noms distinctifs dont les valeurs proviennent uniquement du jeu de caractères Latin-1 . Pour créer des noms distinctifs avec des caractères de l'ensemble, vous devez d'abord créer un certificat avec un nom distinctif créé en UTF-8 à l'aide de UNIX avec le codage UTF-8 activé ou avec l'interface graphique **strmqikm** . Vous devez ensuite créer une règle à partir d'une plateforme UNIX avec le codage UTF-8 activé ou utiliser le plug-in Advanced Message Security dans IBM MQ.

Concepts associés

«Noms distinctifs d'expéditeur dans AMS», à la page 654
Les noms distinctifs d'expéditeur identifient les utilisateurs autorisés à placer des messages dans une file d'attente. Un expéditeur utilise son certificat pour signer un message, avant de le placer dans une file d'attente.

Attributs de stratégie de sécurité dans AMS

Vous pouvez utiliser Advanced Message Security pour sélectionner un algorithme ou une méthode spécifique afin de protéger les données.

Une règle de sécurité est un objet conceptuel qui décrit la façon dont un message est chiffré et signé de manière cryptographique.

Tableau 103. Attributs de stratégie de sécurité dans AMS

Attribut	**Description**
Nom de la règle	Nom unique de la règle d'un gestionnaire de files d'attente.
Algorithme de signature	Algorithme de cryptographie utilisé pour signer les messages avant l'envoi.
Algorithme de chiffrement	Algorithme de cryptographie utilisé pour chiffrer les messages avant leur envoi.
Liste des destinataires	Liste des noms distinctifs (DN) de certificats des destinataires potentiels d'un message.
Liste de contrôle Nom distinctif de signature	Liste des noms distinctifs de signature à valider lors de l'extraction de message.

Dans Advanced Message Security, les messages sont chiffrés avec une clé symétrique et la clé symétrique est chiffrée avec les clés publiques des destinataires. Les clés publiques sont chiffrées avec l'algorithme RSA, avec des clés d'une longueur effective jusqu'à 2048 bits. Le chiffrement de la clé asymétrique dépend de la longueur de la clé de certificat.

Les algorithmes de clé symétrique pris en charge sont les suivants:

- RC2
- DES
- 3DES
- AES128
- AES256

Advanced Message Security prend également en charge les fonctions de hachage cryptographique suivantes:

- MD5
- SHA-1
- SHA-2 Famille :
  - SHA256
  - SHA384 (longueur de clé minimale acceptable-768 bits)
  - SHA512 (longueur de clé minimale acceptable-768 bits)

Remarque : La qualité de protection utilisée pour les fonctions d'insertion et d'obtention de message doit correspondre. S'il existe une non-concordance de la qualité de protection de la règle entre la file d'attente et le message dans la file d'attente, le message n'est pas accepté et est envoyé à la file d'attente de traitement des erreurs. Cette règle s'applique aux files d'attente locales et éloignées.

Qualité de protection

Les règles de protection des données Advanced Message Security impliquent une qualité de protection (QOP).

Les trois niveaux de qualité de protection dans Advanced Message Security sont complétés par un quatrième niveau dans IBM MQ 9.0 et les versions ultérieures, et dépendent tous des algorithmes de cryptographie utilisés pour signer et chiffrer le message:

- Les messages de confidentialité placés dans la file d'attente doivent être signés et chiffrés.
- Intégrité-Les messages placés dans la file d'attente doivent être signés par l'expéditeur.

- Confidentialité-les messages placés dans la file d'attente doivent être chiffrés. Pour plus d'informations, voir «Qualités de protection disponibles avec AMS», à la page 581
- Aucune-aucune protection des données n'est applicable.

Une règle qui stipule que les messages doivent être signés lorsqu'ils sont placés dans une file d'attente possède un QOP INTEGRITY. Une QOP d'INTEGRITY signifie qu'une règle stipule un algorithme de signature, mais pas un algorithme de chiffrement. Les messages protégés contre l'intégrité sont également appelés "SIGNED".

Une règle qui stipule que les messages doivent être signés et chiffrés lorsqu'ils sont placés dans une file d'attente a un QOP de type PRIVACY. Une QOP de PRIVACY signifie que lorsqu'une règle stipule un algorithme de signature et un algorithme de chiffrement. Les messages protégés par la protection de la vie privée sont également appelés "SCELLÉS".

Une règle qui stipule que les messages doivent être chiffrés lorsqu'ils sont placés dans une file d'attente a une qualité de protection des données (QOP) de type CONFIDENTIALITÉ. Une QOP de CONFIDENTIALITÉ signifie qu'une règle spécifie un algorithme de chiffrement.

Une règle qui ne stipule pas d'algorithme de signature ou de chiffrement a un QOP de NONE. Advanced Message Security ne fournit pas de protection des données pour les files d'attente dont la règle est associée à la valeur NONE.

Gestion des politiques de sécurité

Une règle de sécurité est un objet conceptuel qui décrit la façon dont un message est chiffré et signé de manière cryptographique.

L'emplacement à partir duquel toutes les tâches d'administration liées aux règles de sécurité sont exécutées dépend de la plateforme que vous utilisez.

- ULW Sous UNIXet Windows, vous utilisez les commandes DELETE POLICY, DISPLAY POLICYet SET POLICY (ou des commandes PCF équivalentes) pour gérer vos règles de sécurité.
  - UNIX Sous UNIX, les tâches d'administration peuvent être exécutées à partir de *MQ_INSTALLATION_PATH*/bin.
  - Windows Sur les plateformes Windows , les tâches d'administration peuvent être exécutées à partir de n'importe quel emplacement car la variable d'environnement PATH est mise à jour lors de l'installation.
- IBM i Sous IBM i, les commandes DSPMQMSPL, SETMQMSPLet WRKMQMSPL sont installées dans la bibliothèque système QSYS pour la langue principale du système lorsque IBM MQ est installé.

  Des versions de langue nationale supplémentaires sont installées dans les bibliothèques QSYS29xx en fonction du chargement des fonctions de langue. Par exemple, une machine avec l'anglais américain comme langue principale et le coréen comme langue secondaire a les commandes d'anglais américain installées dans QSYS et le chargement de la langue secondaire coréenne dans QSYS2962 comme 2962 est le chargement de la langue pour le coréen.
- z/OS Sous z/OS, les commandes d'administration sont exécutées à l'aide de l'utilitaire de règles de sécurité des messages (CSQ0UTIL). Lorsque des règles sont créées, modifiées ou supprimées dans z/OS, les modifications ne sont pas reconnues par Advanced Message Security tant que le gestionnaire de files d'attente n'est pas arrêté et redémarré ou que la commande z/OS MODIFY n'est pas utilisée pour actualiser la configuration des règles Advanced Message Security . Exemple :

```
F <qmgr ssid>AMSM,REFRESH POLICY
```

Tâches associées

«Création de stratégies de sécurité dans AMS», à la page 659
Les stratégies de sécurité définissent la façon dont un message est protégé lorsque le message est inséré, ou la façon dont un message doit avoir été protégé lorsqu'un message est reçu.

«Modification des règles de sécurité dans AMS», à la page 660
Vous pouvez utiliser Advanced Message Security pour modifier les détails des règles de sécurité que vous avez déjà définies.

«Affichage et vidage des règles de sécurité dans AMS», à la page 660
La commande **dspmqspl** permet d'afficher la liste de toutes les règles de sécurité ou les détails d'une règle nommée en fonction des paramètres de ligne de commande que vous indiquez.

«Suppression de règles de sécurité dans AMS», à la page 662
Pour supprimer des règles de sécurité dans Advanced Message Security, vous devez utiliser la commande `setmqspl` .

fonctionnementAdvanced Message Security

Référence associée

L'utilitaire de règles de sécurité des messages (CSQ0UTIL)

Création de stratégies de sécurité dans AMS

Les stratégies de sécurité définissent la façon dont un message est protégé lorsque le message est inséré, ou la façon dont un message doit avoir été protégé lorsqu'un message est reçu.

Avant de commencer

Certaines conditions d'entrée doivent être remplies lors de la création de règles de sécurité:

- Il doit être en cours d'exécution.
- Le nom d'une règle de sécurité doit respecter les règles de dénomination des objets IBM MQ.
- Vous devez disposer des droits nécessaires pour vous connecter au gestionnaire de files d'attente et créer une règle de sécurité:
  - z/OS Sous z/OS, accordez les droits documentés dans L'utilitaire de règles de sécurité des messages (CSQ0UTIL).
  - Multi Sur les autres plateformes que z/OS, vous devez accorder les droits + connect, + inq et + chg nécessaires à l'aide de la commande setmqaut .

  Pour plus d'informations sur la configuration de la sécurité, voir «Configuration de la sécurité», à la page 130.
- z/OS Sous z/OS, vérifiez que les objets système requis ont été définis conformément aux définitions dans CSQ4INSM.

Exemple

Voici un exemple de création d'une règle sur le gestionnaire de files d'attente QMGR. La règle spécifie que les messages doivent être signés à l'aide de l'algorithme SHA256 et chiffrés à l'aide de l'algorithme AES256 pour les certificats avec le nom distinctif: CN=joe, O=IBM, C=US et DN: CN=jane, O=IBM, C = US. Cette règle est associée à `MY.QUEUE`:

```
setmqspl -m QMGR -p MY.QUEUE -s SHA256 -e AES256 -r CN=joe,O=IBM,C=US -r CN=jane,O=IBM,C=US
```

Voici un exemple de création de règles sur le gestionnaire de files d'attente QMGR. La règle indique que les messages doivent être chiffrés à l'aide de l'algorithme 3DES pour les certificats avec DN: CN=john, O=IBM, C=US et CN=jeff, O=IBM, C=US et signés avec l'algorithme SHA256 pour les certificats avec DN: CN=phil, O=IBM, C=US

```
setmqspl -m QMGR -p MY.OTHER.QUEUE -s SHA256 -e 3DES -r CN=john,O=IBM,C=US -r
CN=jeff,O=IBM,C=US -a CN=phil,O=IBM,C=US
```

Remarque :

- La qualité de la protection utilisée pour l'insertion et l'extraction de message doit correspondre. Si la qualité de protection de la règle définie pour le message est plus faible que celle définie pour une file d'attente, le message est envoyé à la file d'attente de traitement des erreurs. Cette règle est valide pour les files d'attente locales et éloignées.

Référence associée

Liste complète des attributs de la commande setmqspl

Modification des règles de sécurité dans AMS

Vous pouvez utiliser Advanced Message Security pour modifier les détails des règles de sécurité que vous avez déjà définies.

Avant de commencer

- Le gestionnaire de files d'attente sur lequel vous souhaitez travailler doit être en cours d'exécution.
- Vous devez disposer des droits nécessaires pour vous connecter au gestionnaire de files d'attente et créer une règle de sécurité.
  - z/OS Sous z/OS, accordez les droits documentés dans L'utilitaire de règles de sécurité des messages (CSQ0UTIL).
  - Multi Sur les autres plateformes que z/OS, vous devez accorder les droits + connect, + inq et + chg nécessaires à l'aide de la commande setmqaut .

  Pour plus d'informations sur la configuration de la sécurité, voir «Configuration de la sécurité», à la page 130.

Pourquoi et quand exécuter cette tâche

Pour modifier les règles de sécurité, appliquez la commande `setmqspl` à une règle existante fournissant de nouveaux attributs.

Exemple

Voici un exemple de création d'une règle nommée MYQUEUE sur un gestionnaire de files d'attente nommé QMGR, spécifiant que les messages doivent être chiffrés à l'aide de l'algorithme 3DES pour les auteurs (**-a**) ayant des certificats avec le nom distinctif CN=alice, O=IBM, C=US et signés avec l'algorithme SHA256 pour les destinataires (**-r**) ayant des certificats avec le nom distinctif CN=jeff, O=IBM, C = US.

```
setmqspl -m QMGR -p MYQUEUE -e 3DES -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Pour modifier cette règle, exécutez la commande `setmqspl` avec tous les attributs de l'exemple en modifiant uniquement les valeurs que vous souhaitez modifier. Dans cet exemple, la règle créée précédemment est associée à une nouvelle file d'attente et son algorithme de chiffrement est remplacé par AES256:

```
setmqspl -m QMGR -p MYQUEUE -e AES256 -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Référence associée

setmqspl (définition de la règle de sécurité)

Affichage et vidage des règles de sécurité dans AMS

La commande **dspmqspl** permet d'afficher la liste de toutes les règles de sécurité ou les détails d'une règle nommée en fonction des paramètres de ligne de commande que vous indiquez.

Avant de commencer

- Pour afficher les détails des règles de sécurité, le gestionnaire de files d'attente doit exister et être en cours d'exécution.

- Vous devez disposer des droits nécessaires pour vous connecter au gestionnaire de files d'attente et créer une règle de sécurité.
  - z/OS Sous z/OS, accordez les droits documentés dans L'utilitaire de règles de sécurité des messages (CSQ0UTIL).
  - Multi Sur les autres plateformes que z/OS, vous devez accorder les droits + connect, + inq et + chg nécessaires à l'aide de la commande setmqaut .

  Pour plus d'informations sur la configuration de la sécurité, voir «Configuration de la sécurité», à la page 130.

Pourquoi et quand exécuter cette tâche

Voici la liste des indicateurs de commande **dspmqspl** :

*Tableau 104. Indicateurs de commande **dspmqspl** .*

Indicateur de commande	Explication
-m	Nom du gestionnaire de files d'attente (obligatoire).
-p	Nom de la règle.
-export	L'ajout de cet indicateur génère une sortie qui peut facilement être appliquée à un autre gestionnaire de files d'attente.

Exemple

L'exemple suivant montre comment créer deux règles de sécurité pour `venus.queue.manager`:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s sha256 -a "CN=signer1,O=IBM,C=US" -e NONE
setmqspl -m venus.queue.manager -p AMS_POL_06_THREE -s sha256 -a "CN=another signer,O=IBM,C=US"
-e NONE
```

Cet exemple illustre une commande qui affiche les détails de toutes les règles définies pour `venus.queue.manager` et la sortie qu'elle génère:

```
dspmqspl -m venus.queue.manager

Policy Details:
Policy name: AMS_POL_04_ONE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNs:
  CN=signer1,O=IBM,C=US
Recipient DNs: -
Toleration: 0
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

Cet exemple illustre une commande qui affiche les détails d'une règle de sécurité sélectionnée définie pour `venus.queue.manager` et la sortie qu'elle génère:

```
dspmqspl -m venus.queue.manager -p AMS_POL_06_THREE
```

```
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

Dans l'exemple suivant, nous créons d'abord une règle de sécurité, puis nous exportons la règle à l'aide de l'indicateur **-export** :

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s SHA256 -a "CN=signer1,O=IBM,C=US" -e NONE

dspmqspl -m venus.queue.manager -export
```

z/OS Sous z/OS, les informations de règle exportées sont écrites par CSQ0UTIL dans la définition de données EXPORT.

Multi Sur les plateformes autres que z/OS, redirigez la sortie vers un fichier, par exemple:

```
dspmqspl -m venus.queue.manager -export > policies.[bat|sh]
```

Pour importer une règle de sécurité:

- Windows Sous Windows, exécutez `policies.bat`.
- UNIX Sous UNIX :
  1. Connectez-vous en tant qu'utilisateur appartenant au groupe d'administration `mqm` IBM MQ .
  2. Exécutez `. policies.sh`.
- z/OS Sous z/OS , utilisez l'utilitaire CSQ0UTIL , en indiquant à SYSIN le fichier contenant les informations de règle exportées.

Référence associée

Liste complète des attributs de la commande dspmqspl

Suppression de règles de sécurité dans AMS

Pour supprimer des règles de sécurité dans Advanced Message Security, vous devez utiliser la commande `setmqspl` .

Avant de commencer

Certaines conditions d'entrée doivent être remplies lors de la gestion des règles de sécurité:

- Il doit être en cours d'exécution.
- Vous devez disposer des droits nécessaires pour vous connecter au gestionnaire de files d'attente et créer une règle de sécurité.
  - z/OS Sous z/OS, accordez les droits documentés dans L'utilitaire de règles de sécurité des messages (CSQ0UTIL).
  - Multi Sur les autres plateformes que z/OS, vous devez accorder les droits + connect, + inq et + chg nécessaires à l'aide de la commande setmqaut .

  Pour plus d'informations sur la configuration de la sécurité, voir «Configuration de la sécurité», à la page 130.

Pourquoi et quand exécuter cette tâche

Utilisez la commande **setmqspl** avec l'option **-remove** .

Exemple

Voici un exemple de suppression d'une règle:

```
setmqspl -m QMGR -remove -p MY.OTHER.QUEUE
```

Référence associée

Liste complète des attributs de la commande setmqspl

Protection des files d'attente système dans AMS

Les files d'attente système permettent la communication entre IBM MQ et ses applications auxiliaires. Chaque fois qu'un gestionnaire de files d'attente est créé, une file d'attente système est également créée pour stocker les messages et les données internes IBM MQ . Vous pouvez protéger les files d'attente système avec Advanced Message Security afin que seuls les utilisateurs autorisés puissent y accéder ou les déchiffrer.

La protection des files d'attente système suit le même modèle que la protection des files d'attente standard. Voir «Création de stratégies de sécurité dans AMS», à la page 659.

Windows Pour utiliser la protection de file d'attente système sous Windows, copiez le fichier `keystore.conf` dans le répertoire suivant:

```
c:\Documents and Settings\Default User\.mqs\keystore.conf
```

z/OS Sous z/OS, pour protéger `SYSTEM.ADMIN.COMMAND.QUEUE`, le serveur de commandes doit avoir accès à `keystore` et à `keystore.conf`, qui contiennent des clés et une configuration permettant au serveur de commandes d'accéder aux clés et aux certificats. Toutes les modifications apportées à la règle de sécurité de `SYSTEM.ADMIN.COMMAND.QUEUE` nécessitent le redémarrage du serveur de commandes.

Tous les messages envoyés et reçus à partir de la file d'attente de commandes sont signés ou signés et chiffrés en fonction des paramètres de règle. Si un administrateur définit des signataires autorisés, les messages de commande qui ne passent pas la vérification du nom distinctif (DN) du signataire ne sont pas exécutés par le serveur de commandes et ne sont pas acheminés vers la file d'attente de traitement d'erreurs Advanced Message Security . Les messages envoyés en tant que réponses à des files d'attente dynamiques temporaires IBM MQ Explorer ne sont pas protégés par AMS.

Les règles de sécurité n'ont pas d'effet sur les files d'attente SYSTEM suivantes:

- SYSTEM.ADMIN.ACCOUNTING.QUEUE
- SYSTEM.ADMIN.ACTIVITY.QUEUE
- SYSTEM.ADMIN.CHANNEL.EVENT
- SYSTEM.ADMIN.COMMAND.EVENT
- z/OS SYSTEM.ADMIN.COMMAND.QUEUE
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.LOGGER.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.PUBSUB.EVENT
- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.STATISTICS.QUEUE
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE

- SYSTEM.AUTH.DATA.QUEUE
- SYSTEM.BROKER.ADMIN.STREAM
- z/OS SYSTEM.BROKER.CLIENTS.DATA
- SYSTEM.BROKER.CONTROL.QUEUE
- SYSTEM.BROKER.DEFAULT.STREAM
- SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS
- z/OS SYSTEM.BROKER.SUBSCRIPTIONS.DATA
- SYSTEM.CHANNEL.INITQ
- SYSTEM.CHANNEL.SYNCQ
- z/OS SYSTEM.CHLAUTH.DATA.QUEUE
- SYSTEM.CICS.INITIATION.QUEUE
- SYSTEM.CLUSTER.COMMAND.QUEUE
- SYSTEM.CLUSTER.HISTORY.QUEUE
- SYSTEM.CLUSTER.REPOSITORY.QUEUE
- SYSTEM.CLUSTER.TRANSMIT.QUEUE
- z/OS SYSTEM.COMMAND.INPUT
- z/OS SYSTEM.DDELAY.LOCAL.QUEUE
- SYSTEM.DEAD.LETTER.QUEUE
- SYSTEM.DURABLE.SUBSCRIBER.QUEUE
- SYSTEM.HIERARCHY.STATE
- SYSTEM.INTER.QMGR.CONTROL
- SYSTEM.INTER.QMGR.FANREQ
- SYSTEM.INTER.QMGR.PUBS
- SYSTEM.INTERNAL.REPLY.QUEUE
- z/OS SYSTEM.JMS.PS.STATUS.QUEUE
- z/OS SYSTEM.JMS.REPORT.QUEUE
- SYSTEM.PENDING.DATA.QUEUE
- SYSTEM.PROTECTION.ERROR.QUEUE
- SYSTEM.PROTECTION.POLICY.QUEUE
- z/OS SYSTEM.QSG.CHANNEL.SYNCQ
- z/OS SYSTEM.QSG.TRANSMIT.QUEUE
- z/OS SYSTEM.QSG.UR.RESOLUTION.QUEUE
- SYSTEM.RETAINED.PUB.QUEUE
- z/OS SYSTEM.RETAINED.PUB.QUEUE
- SYSTEM.SELECTION.EVALUATION.QUEUE
- SYSTEM.SELECTION.VALIDATION.QUEUE

Octroi de droits OAM

Les droits d'accès aux fichiers autorisent tous les utilisateurs à exécuter les commandes `setmqspl` et `dspmqspl` . Toutefois, Advanced Message Security s'appuie sur le gestionnaire des droits d'accès aux

objets (OAM) et toute tentative d'exécution de ces commandes par un utilisateur qui n'appartient pas au groupe mqm, qui est le groupe d'administration IBM MQ , ou qui ne dispose pas des droits permettant de lire les paramètres de règles de sécurité accordés, génère une erreur.

Procédure

Pour accorder les droits nécessaires à un utilisateur, exécutez:

```
setmqaut -m SOME.QUEUE.MANAGER -t qmgr -p SOME.USER +connect +inq
setmqaut -m SOME.QUEUE.MANAGER -t queue -n SYSTEM.PROTECTION.POLICY.QUEUE -p SOME.USER +browse
+put
setmqaut -m SOME.QUEUE.MANAGER -t queue -n SYSTEM.PROTECTION.ERROR.QUEUE -p SOME.USER +put
```

Remarque : Vous n'avez besoin de définir ces droits OAM que si vous prévoyez de connecter des clients au gestionnaire de files d'attente à l'aide de Advanced Message Security 7.0.1.

Avertissement : Parcourez les droits d'accès à SYSTEM.PROTECTION.POLICY.QUEUE n'est pas obligatoire dans toutes les situations. IBM MQ optimise les performances en mettant en cache les règles de sorte que vous n'ayez pas à parcourir les enregistrements pour obtenir des détails sur les règles dans SYSTEM.PROTECTION.POLICY.QUEUE dans tous les cas.

IBM MQ ne met pas en cache toutes les règles disponibles. S'il existe un nombre élevé de règles, IBM MQ met en cache un nombre limité de règles. Par conséquent, si le nombre de règles définies pour le gestionnaire de files d'attente est faible, il n'est pas nécessaire de fournir l'option de navigation à SYSTEM.PROTECTION.POLICY.QUEUE.

Toutefois, vous devez accorder le droit de navigation à cette file d'attente, au cas où un nombre élevé de règles serait défini, ou si vous utilisez d'anciens clients. SYSTEM.PROTECTION.ERROR.QUEUE est utilisé pour placer les messages d'erreur générés par le code AMS. Le droit d'insertion sur cette file d'attente est vérifié uniquement lorsque vous tentez d'insérer un message d'erreur dans la file d'attente. Votre droit d'insertion sur la file d'attente n'est pas vérifié lorsque vous tentez d'insérer ou d'extraire un message d'une file d'attente protégée AMS.

Octroi de droits de sécurité

Lors de l'utilisation de la sécurité des ressources de commande, vous devez définir des droits pour permettre à Advanced Message Security de fonctionner. Cette rubrique utilise les commandes RACF dans les exemples. Si votre entreprise utilise un gestionnaire de sécurité externe (ESM) différent, vous devez utiliser les commandes équivalentes pour ce gestionnaire.

L'octroi de droits de sécurité comporte trois aspects:

- «Espace adresse AMSM», à la page 666
- «CSQ0UTIL», à la page 666
- «Utilisation de files d'attente pour lesquelles une règle Advanced Message Security est définie», à la page 666

Remarques : Les exemples de commande utilisent les variables suivantes.

1. *QMgrName* -nom du gestionnaire de files d'attente.

   z/OS Sous z/OS, cette valeur peut également être le nom d'un groupe de partage de files d'attente.
2. *username* -il peut s'agir d'un nom de groupe.
3. Les exemples montrent la classe MQQUEUE. Il peut également s'agir de MXQUEUE, GMQQUEUE ou GMXQUEUE. Pour plus d'informations, voir «Profils pour la sécurité de la file d'attente», à la page 203.

De plus, si le profil existe déjà, vous n'avez pas besoin de la commande RDEFINE.

Espace adresse AMSM

Vous devez fournir une sécurité IBM MQ au nom d'utilisateur sous lequel l'espace adresse Advanced Message Security s'exécute.

- Pour la connexion par lots au gestionnaire de files d'attente, émettez

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
           PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Permet d'accéder à SYSTEM.PROTECTION.POLICY.QUEUE, problème:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
           PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

CSQ0UTIL

L'utilitaire qui permet aux utilisateurs d'exécuter les commandes **setmqspl** et **dspmqspl** requiert les droits suivants, où le nom d'utilisateur correspond à l'ID utilisateur du travail:

- Pour la connexion par lots au gestionnaire de files d'attente, émettez:

```
  RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
            PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Permet d'accéder à SYSTEM.PROTECTION.POLICY.QUEUE, requis pour la commande **setmqpol** , exécutez:

```
  RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
             PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(ALTER)
```

- Permet d'accéder à SYSTEM.PROTECTION.POLICY.QUEUE, requis pour la commande **dspmqpol** , exécutez:

```
  RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
              PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

Utilisation de files d'attente pour lesquelles une règle Advanced Message Security est définie

Lorsqu'une application utilise des files d'attente pour lesquelles une règle est définie, elle requiert des droits supplémentaires pour permettre à Advanced Message Security de protéger les messages.

L'application requiert:

- Accès en lecture à SYSTEM.PROTECTION.POLICY.QUEUE. Pour ce faire, exécutez la commande suivante:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
          PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

- Placez l'accès à SYSTEM.PROTECTION.ERROR.QUEUE. Pour ce faire, exécutez la commande suivante:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE UACC(NONE)
           PERMIT QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

IBM i Configuration des certificats et du fichier de configuration du magasin de clés sous IBM i

Votre première tâche lors de la configuration de la protection Advanced Message Security consiste à créer un certificat et à l'associer à votre environnement. L'association est configurée via un fichier stocké dans le système de fichiers intégré (IFS).

Procédure

1. Pour créer un certificat autosigné à l'aide des outils OpenSSL fournis avec IBM i, exécutez la commande suivante à partir de QShell:

   ```
   /QOpenSys/usr/bin/openssl req -x509 -newkey rsa:2048 -keyout
   $HOME/private.pem -out $HOME/mycert.pem -nodes -days 365
   ```

   La commande vous invite à indiquer divers attributs de nom distinctif pour un nouveau certificat autosigné, notamment:

   - Nom usuel (CN =)
   - Organisation (O =)
   - Pays (C =)

   Cela crée une clé privée non chiffrée et un certificat correspondant, au format PEM (Privacy Enhanced Mail).

   Par souci de simplicité, entrez simplement des valeurs pour le nom usuel, l'organisation et le pays. Ces attributs et valeurs sont importants lors de la création d'une règle.

   Des invites et des attributs supplémentaires peuvent être personnalisés en spécifiant un fichier de configuration openssl personnalisé sur la ligne de commande avec le paramètre **-config** . Pour plus de détails sur la syntaxe du fichier de configuration, voir la documentation OpenSSL .

   Par exemple, la commande suivante ajoute des extensions de certificat X.509 v3 supplémentaires:

   ```
   /QOpenSys/usr/bin/openssl req -x509 -newkey rsa:2048
   -keyout $HOME/private.pem -out $HOME/mycert.pem -nodes -days 365 -config myconfig.cnf
   ```

   où myconfig.cnf est un fichier de flux ASCII qui contient les éléments suivants:

   ```
   [req]
   distinguished_name = req_distinguished_name
   x509_extensions = myextensions

   [req_distinguished_name]
   countryName = Country Name (2 letter code)
   countryName_default = GB
   stateOrProvinceName = State or Province Name (full name)
   stateOrProvinceName_default = Hants
   localityName = Locality Name (eg, city)
   localityName_default = Hursley
   organizationName = Organization Name (eg, company)
   organizationName_default = IBM United Kingdom
   organizationalUnitName = Organizational Unit Name (eg, department)
   organizationalUnitName_default = IBM MQ Development
   commonName = Common Name (eg, Your Name)

   [myextensions]
   keyUsage = digitalSignature,nonRepudiation,dataEncipherment,keyEncipherment
   extendedKeyUsage = emailProtection
   ```

2. AMS requiert que le certificat et la clé privée soient conservés dans le même fichier. Exécutez la commande suivante pour ce faire:

   ```
   cat $HOME/mycert.pem >> $HOME/private.pem
   ```

Le fichier `private.pem` dans $HOME contient désormais une clé privée et un certificat correspondants, tandis que le fichier `mycert.pem` contient tous les certificats publics pour lesquels vous pouvez chiffrer les messages et valider les signatures.

Les deux fichiers doivent être associés à votre environnement en créant un fichier de configuration de magasin de clés, `keystore.conf`, dans votre emplacement par défaut.

Par défaut, AMS recherche la configuration du magasin de clés dans un sous-répertoire `.mqs` de votre répertoire de base.

3. Dans QShell, créez le fichier `keystore.conf` :

```
mkdir -p $HOME/.mqs
echo "pem.private = $HOME/private.pem" > $HOME/.mqs/keystore.conf
echo "pem.public = $HOME/mycert.pem" >> $HOME/.mqs/keystore.conf
echo "pem.password = unused" >> $HOME/.mqs/keystore.conf
```

IBM i *Création d'une règle sur IBM i*

Avant de créer une règle, vous devez créer une file d'attente pour stocker les messages protégés.

Procédure

1. A l'invite de ligne de commande, entrez ;

```
CRTMQMQ QNAME(PROTECTED) QTYPE(*LCL) MQMNAME (mqmname)
```

où mqmname est le nom de votre gestionnaire de files d'attente.

Utilisez la commande DSPMQM pour vérifier que le gestionnaire de files d'attente est capable d'utiliser des règles de sécurité. Vérifiez que **Security Policy Capability** affiche **YES*.

La règle la plus simple que vous pouvez définir est une règle d'intégrité, qui est obtenue en créant une règle avec un algorithme de signature numérique mais pas d'algorithme de chiffrement.

Les messages sont signés mais non chiffrés. Si les messages doivent être chiffrés, vous devez spécifier un algorithme de chiffrement et un ou plusieurs destinataires de message prévus.

Un certificat dans le magasin de clés public d'un destinataire de message est identifié par un nom distinctif.

2. Affichez les noms distinctifs des certificats dans le magasin de clés public, `mycert.pem` dans $HOME, à l'aide de la commande suivante dans QShell:

```
/QOpenSys/usr/bin/openssl x509 -in $HOME/mycert.pem -noout -subject -nameopt RFC2253
```

Vous devez entrer le nom distinctif comme destinataire prévu et le nom de la règle doit correspondre au nom de la file d'attente à protéger.

3. Dans une invite de commande CL, entrez, par exemple:

```
SETMQMSPL POLICY(PROTECTED) MQMNAME (mqmname)SIGNALG(*SHA256) ENCALG(*AES256) RECIP('CN=..,
O=.., C=..')
```

où mqmname est le nom de votre gestionnaire de files d'attente.

Une fois la règle créée, tous les messages insérés, consultés ou supprimés de façon destructive via ce nom de file d'attente sont soumis à la règle AMS .

Référence associée

Afficher le gestionnaire de files d'attente de messages (DSPMQM)

Définir une règle de sécurité MQM (SETMQMSPL)

IBM i *Test de la règle sur IBM i*

Utilisez les exemples d'application fournis avec le produit pour tester vos stratégies de sécurité.

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser les exemples d'application fournis avec IBM MQ , tels que AMQSPUT4, AMQSGET4, AMQSGBR4et des outils tels que WRKMQMMSG pour insérer, parcourir et extraire des messages à l'aide du nom de file d'attente PROTECTED.

Si tout a été configuré correctement, le comportement de l'application ne doit pas être différent de celui d'une file d'attente non protégée pour cet utilisateur.

Un utilisateur non configuré pour Advanced Message Securityou un utilisateur qui ne dispose pas de la clé privée requise pour déchiffrer le message ne pourra toutefois pas afficher le message. L'utilisateur reçoit le code achèvement RCFAIL, équivalent à MQCC_FAILED (2) et le code anomalie RC2063 (MQRC_SECURITY_ERROR).

Pour vérifier que la protection AMS est activée, placez des messages de test dans la file d'attente PROTECTED, par exemple à l'aide de AMQSPUT0. Vous pouvez ensuite créer une file d'attente alias pour parcourir les données protégées brutes alors qu'elles sont au repos.

Procédure

Pour accorder les droits nécessaires à un utilisateur, exécutez:

```
CRTMQMQ QNAME(ALIAS) QTYPE(*ALS) TGTQNAME(PROTECTED) MQMNAME(yourqm)
```

L'exploration à l'aide du nom de file d'attente ALIAS, par exemple à l'aide de AMQSBCG4 ou WRKMQMMSG, doit révéler des messages `scrambled` plus volumineux dans lesquels une exploration de la file d'attente PROTECTED affiche des messages en clair.

Les messages brouillés sont visibles, mais le texte en clair d'origine n'est pas déchiffrable à l'aide de la file d'attente ALIAS, car il n'existe pas de règle permettant à AMS d'imposer la mise en correspondance de ce nom. Par conséquent, les données protégées brutes sont renvoyées.

Référence associée

Définir une règle de sécurité MQM (SETMQMSPL)
Gestion des messages MQ (WRKMQMMSG)

Evénements de commande et de configuration

Avec Advanced Message Security, vous pouvez générer des messages d'événement de commande et de configuration, qui peuvent être consignés et servir d'enregistrement des changements de règles à des fins d'audit.

Les événements de commande et de configuration générés par IBM MQ sont des messages au format PCF envoyés aux files d'attente dédiées sur le gestionnaire de files d'attente où l'événement se produit.

Les messages d'événements de configuration sont envoyés à SYSTEM.ADMIN.CONFIG.EVENT EVENT.

Les messages d'événements de commande sont envoyés à SYSTEM.ADMIN.COMMAND.EVENT EVENT.

Les événements sont générés indépendamment des outils que vous utilisez pour gérer les règles de sécurité Advanced Message Security .

Dans Advanced Message Security, il existe quatre types d'événements générés par différentes actions sur les règles de sécurité:

- «Création de stratégies de sécurité dans AMS», à la page 659, qui génère deux messages d'événement IBM MQ :
  - Un événement de configuration

  - Un événement de commande
- «Modification des règles de sécurité dans AMS», à la page 660, qui génère trois messages d'événement IBM MQ :
  - Evénement de configuration contenant d'anciennes valeurs de règles de sécurité
  - Evénement de configuration contenant de nouvelles valeurs de règle de sécurité
  - Un événement de commande
- «Affichage et vidage des règles de sécurité dans AMS», à la page 660, qui génère un message d'événement IBM MQ :
  - Un événement de commande
- «Suppression de règles de sécurité dans AMS», à la page 662, qui génère deux messages d'événement IBM MQ :
  - Un événement de configuration
  - Un événement de commande

Activation et désactivation de la journalisation des événements

Vous pouvez contrôler les événements de commande et de configuration à l'aide des attributs de gestionnaire de files d'attente **CONFIGEV** et **CMDEV**. Pour activer ces événements, définissez l'attribut de gestionnaire de files d'attente approprié sur ENABLED. Pour désactiver ces événements, définissez l'attribut de gestionnaire de files d'attente approprié sur DISABLED.

Procédure

Evénements de configuration

Pour activer les événements de configuration, définissez **CONFIGEV** sur ENABLED. Pour désactiver les événements de configuration, définissez **CONFIGEV** sur DISABLED. Par exemple, vous pouvez activer des événements de configuration à l'aide de la commande MQSC suivante:

```
ALTER QMGR CONFIGEV (ENABLED)
```

Evénements Commande

Pour activer les événements de commande, définissez **CMDEV** sur ENABLED. Pour activer les événements de commande pour les commandes à l'exception des commandes **DISPLAY MQSC** et des commandes Inquire PCF, définissez **CMDEV** sur NODISPLAY. Pour désactiver les événements de commande, définissez **CMDEV** sur DISABLED. Par exemple, vous pouvez activer des événements de commande à l'aide de la commande MQSC suivante:

```
ALTER QMGR CMDEV (ENABLED)
```

Tâches associées

Contrôle des événements de configuration, de commande et de consignateur dans IBM MQ

Format de message d'événement de commande

Le message d'événement de commande se compose de la structure MQCFH et des paramètres PCF qui la suivent.

Voici les valeurs MQCFH sélectionnées:

```
Type = MQCFT_EVENT;
Command = MQCMD_COMMAND_EVENT;
MsgSeqNumber = 1;
Control = MQCFC_LAST;
ParameterCount = 2;
CompCode = MQCC_WARNING;
Reason = MQRC_COMMAND_PCF;
```

Remarque : La valeur de ParameterCount est deux car il existe toujours deux paramètres de type MQCFGR (groupe). Chaque groupe est constitué de paramètres appropriés. Les données d'événement se composent de deux groupes, CommandContext et CommandData.

CommandContext contient:

EventUserId

Description :	ID utilisateur qui a émis la commande ou l'appel qui a généré l'événement. (Il s'agit du même ID utilisateur que celui utilisé pour vérifier les droits d'émission de la commande ou de l'appel ; pour les commandes reçues d'une file d'attente, il s'agit également de l'ID utilisateur (UserIdentifier) du MD du message de commande).
Identificateur :	MQCACF_EVENT_USER_ID.
Type de données :	MQCFST.
Longueur maximale :	MQ_USER_ID_LENGTH.
Renvoyé:	Toujours.

EventOrigin

Description :	Origine de l'action à l'origine de l'événement.
Identificateur :	MQIACF_EVENT_ORIGIN.
Type de données :	MQCFIN.
Valeurs :	**MQEVO_CONSOLE** Ligne de commande de la console. **MQEVO_MSG** Message de commande du plug-in IBM MQ Explorer .
Renvoyé:	Toujours.

EventQMgr

Description :	Gestionnaire de files d'attente dans lequel la commande ou l'appel a été entré. (Le gestionnaire de files d'attente dans lequel la commande est exécutée et qui génère l'événement se trouve dans le descripteur du message d'événement).
Identificateur :	MQCACF_EVENT_Q_MGR.
Type de données :	MQCFST.
Longueur maximale :	MQ_Q_MGR_NAME_LENGTH.
Renvoyé:	Toujours.

EventAccountingToken

Description :	Pour les commandes reçues sous forme de message (MQEVO_MSG), jeton de comptabilité (AccountingToken) provenant du descripteur de message de commande.
Identificateur :	MQBACF_EVENT_ACCOUNTING_TOKEN.
Type de données :	MQCFBS.
Longueur maximale :	MQ_ACCOUNTING_TOKEN_LENGTH.
Renvoyé:	Uniquement si EventOrigin est MQEVO_MSG.

Données EventIdentity

Description :	Pour les commandes reçues sous forme de message (MQEVO_MSG), données d'identité d'application (donnéesApplIdentity) provenant du descripteur de message de commande.
Identificateur :	MQCACF_EVENT_APPL_IDENTITY.
Type de données :	MQCFST.
Longueur maximale :	MQ_APPL_IDENTITY_DATA_LENGTH.
Renvoyé:	Uniquement si EventOrigin est MQEVO_MSG.

EventApplType

Description :	Pour les commandes reçues sous forme de message (MQEVO_MSG), type d'application (PutApplType) à partir du descripteur de message du message de commande.
Identificateur :	MQIACF_EVENT_APPL_TYPE.
Type de données :	MQCFIN.
Renvoyé:	Uniquement si EventOrigin est MQEVO_MSG.

EventApplName

Description :	Pour les commandes reçues sous forme de message (MQEVO_MSG), nom de l'application (nomPutAppl) à partir du descripteur de message du message de commande.
Identificateur :	MQCACF_EVENT_APPL_NAME.
Type de données :	MQCFST.
Longueur maximale :	MQ_APPL_NAME_LENGTH.
Renvoyé:	Uniquement si EventOrigin est MQEVO_MSG.

EventApplOrigin

Description :	Pour les commandes reçues sous forme de message (MQEVO_MSG), les données d'origine de l'application (donnéesApplOrigin) provenant du descripteur de message de commande.
Identificateur :	MQCACF_EVENT_APPL_ORIGIN.
Type de données :	MQCFST.
Longueur maximale :	MQ_APPL_ORIGIN_DATA_LENGTH.
Renvoyé:	Uniquement si EventOrigin est MQEVO_MSG.

Commande

Description :	Code de la commande.
Identificateur :	MQIACF_COMMAND.
Type de données :	MQCFIN.

Valeurs : **Valeur numérique MQCMD_INQUIRE_PROT_POLICY 205**

Valeur numérique 206 de MQCMD_CREATE_PROT_POLICY

Valeur numérique 207 de MQCMD_DELETE_PROT_POLICY

Valeur numérique 208 de MQCMD_CHANGE_PROT_POLICY

Ils sont définis dans IBM MQ 8.0 `cmqcfc.h`

Renvoyé: Toujours.

CommandData contient des éléments PCF qui composent la commande PCF.

Format de message d'événement de configuration

Les événements de configuration sont des messages PCF au format Advanced Message Security standard.

Les valeurs possibles pour le descripteur de message MQMD se trouvent dans Message d'événement MQMD (descripteur de message).

Voici les valeurs MQMD sélectionnées:

```
Format = MQFMT_EVENT
Peristence = MQPER_PERSISTENCE_AS_Q_DEF
PutApplType = MQAT_QMGR           //for both CLI and command server
```

La mémoire tampon de messages se compose de la structure MQCFH et de la structure de paramètres qui la suit. Les valeurs MQCFH possibles se trouvent dans Message d'événement MQCFH (en-tête PCF).

Voici les valeurs MQCFH sélectionnées:

```
Type = MQCFT_EVENT
Command = MQCMD_CONFIG_EVENT
MsgSeqNumber = 1 or 2          // 2 will be in case of Change Object event
Control = MQCFC_LAST or MQCFC_NOT_LAST     //MQCFC_NOT_LAST will be in case of 1 Change Object
event
ParameterCount = reflects number of PCF parameters following MQCFH
CompCode = MQCC_WARNING
Reason = one of {MQRC_CONFIG_CREATE_OBJECT, MQRC_CONFIG_CHANGE_OBJECT,
MQRC_CONFIG_DELETE_OBJECT}
```

Les paramètres suivants de MQCFH sont:

EventUserID

Description : ID utilisateur qui a émis la commande ou l'appel qui a généré l'événement. (Il s'agit du même ID utilisateur que celui utilisé pour vérifier les droits d'émission de la commande ou de l'appel ; pour les commandes reçues d'une file d'attente, il s'agit également de l'ID utilisateur (UserIdentifier) du MD du message de commande).

Identificateur : **MQCACF_EVENT_USER_ID**

Type de données : MQCFST.

Longueur maximale : MQ_USER_ID_LENGTH.

Renvoyé: Toujours.

SecurityId

Description : Valeur de MQMD.AccountingToken dans le cas d'un message du serveur de commandes ou Windows SID pour la commande locale.

Identificateur : **ID_SÉCURITÉ_ÉVÉNEMENT_MQBACF**

Type de données : MQCBS.

Longueur maximale : MQ_SECURITY_ID_LENGTH.

Renvoyé: Toujours.

EventOrigin

Description : Origine de l'action à l'origine de l'événement.

Identificateur : **MQIACF_EVENT_ORIGIN**

Type de données : MQCFIN.

Valeurs :

MQEVO_CONSOLE
Ligne de commande de la console.

MSG MQEVO_MQ
Message de commande du plug-in IBM MQ Explorer.

Renvoyé: Toujours.

EventQMgr

Description : Gestionnaire de files d'attente dans lequel la commande ou l'appel a été entré. (Le gestionnaire de files d'attente dans lequel la commande est exécutée et qui génère l'événement se trouve dans le descripteur du message d'événement).

Identificateur : **MQCACF_EVENT_Q_MGR**

Type de données : MQCFST

Longueur maximale : MQ_Q_MGR_NAME_LENGTH

Renvoyé: Toujours.

ObjectType

Description : Types d'objet.

Identificateur : **MQIACF_OBJECT_TYPE**

Type de données : MQCFIN

Valeur :

POLITIQUE MQOT_PROT_DE
Règle de protection Advanced Message Security . **1019** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Renvoyé: Toujours.

PolicyName

Description : Nom de la règle Advanced Message Security .

Identificateur : **MQCA_POLICY_NAME**.

Type de données : MQCFST.

Valeur : **2112** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Longueur maximale : MQ_OBJECT_NAME_LENGTH.

Renvoyé: Toujours.

PolicyVersion

Description : Version de la règle Advanced Message Security .

Identificateur : **MQIA_POLICY_VERSION**

Type de données : MQCFIN

Valeur **238** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Renvoyé: Toujours

TolerateFlag

Description : Indicateur de tolérance de la règle Advanced Message Security .

Identificateur : **MQIA_TOLERATE_NON protégé**

Type de données : MQCFIN

Valeur **235** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Renvoyé: Toujours.

SignatureAlgorithm

Description : Algorithme de signature de règle Advanced Message Security .

Identificateur : **Algorithme de signature (MQIA_SIGNATURE_ALGORITHM)**

Type de données : MQCFIN

Valeur : **236** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Renvoyé: Chaque fois qu'un algorithme de signature est défini dans la règle Advanced Message Security

EncryptionAlgorithm

Description : Algorithme de chiffrement de la règle Advanced Message Security .

Identificateur : **Algorithme de chiffrement MQIA_ENCRYPTION_ALGORITHM**

Type de données : MQCFIN

Valeur : **237** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Renvoyé: Chaque fois qu'un algorithme de chiffrement est défini dans la règle IBM MQ

SignerDNs

Description : Sujet DistinguishedName des signataires autorisés.

Identificateur : **MQCA_SIGNER_DN**

Type de données : MQCFSL

Valeur : **2113** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .

Longueur maximale : Nom distinctif de signataire le plus long dans la règle, mais pas plus long que MQ_DISTINGUISHED_NAME_LENGTH

Renvoyé: Chaque fois qu'il est défini dans la règle IBM MQ .

RecipientDNs

Description : Sujet DistinguishedName des signataires autorisés.

Identificateur : **MQCA_RECIPIENT_DN**

Type de données :	MQCFSL
Valeur :	**2114** -Valeur numérique définie dans IBM MQ 8.0 ou dans le fichier `cmqc.h` .
Longueur maximale :	Nom distinctif du destinataire le plus long dans la règle, mais pas MQ_DISTINGUISHED_NAME_LENGTH.
Renvoyé:	Chaque fois qu'il est défini dans la règle IBM MQ .

Remarques

:NONE.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Corporation
Tour Descartes
Armonk, NY 10504-1785
U.S.A.

Pour toute demande d'informations relatives au jeu de caractères codé sur deux octets, contactez le service de propriété intellectuelle IBM ou envoyez vos questions par courrier à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japon

Le paragraphe suivant ne s’applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
Coordinateur d'interopérabilité logicielle, département 49XA
3605 Autoroute 52 N

Rochester, MN 55901
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans le présent document et tous les éléments sous disponibles s'y rapportant sont fournis par IBM conformément aux dispositions du Contrat sur les produits et services IBM, aux Conditions Internationales d'Utilisation de Logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Licence sur les droits d'auteur :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Documentation sur l'interface de programmation

Les informations d'interface de programmation, si elles sont fournies, sont destinées à vous aider à créer un logiciel d'application à utiliser avec ce programme.

Ce manuel contient des informations sur les interfaces de programmation prévues qui permettent au client d'écrire des programmes pour obtenir les services de WebSphere MQ.

Toutefois, lesdites informations peuvent également contenir des données de diagnostic, de modification et d'optimisation. Ces données vous permettent de déboguer votre application.

Important : N'utilisez pas ces informations de diagnostic, de modification et d'optimisation en tant qu'interface de programmation car elles sont susceptibles d'être modifiées.

Marques

IBM, le logo IBM , ibm.com, sont des marques d' IBM Corporation dans de nombreux pays. La liste actualisée de toutes les marques d' IBM est disponible sur la page Web "Copyright and trademark

information"www.ibm.com/legal/copytrade.shtml. Les autres noms de produits et de services peuvent être des marques d'IBM ou d’autres sociétés.

Microsoft et Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans d'autres pays.

UNIX est une marque de The Open Group aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Ce produit inclut des logiciels développés par le projet Eclipse (http://www.eclipse.org/).

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.

Référence :

(1P) P/N: