7.5

Protegendo o IBM WebSphere MQ

Nota

Antes de usar estas informações e o produto que elas suportam, leia as informações em “Avisos” na página 331.

Esta edição se aplica à versão 7 liberação 5 do IBM® WebSphere MQ e a todas as liberações e modificações subsequentes até que seja indicado de outra forma em novas edições.

Ao enviar informações para a IBM, você concede à IBM um direito não exclusivo de usar ou distribuir as informações da maneira que julgar apropriada, sem incorrer em qualquer obrigação para com você

Índice

Segurança

A segurança é uma consideração importante para os desenvolvedores dos aplicativos IBM WebSphere MQ e para os administradores do sistema, configurando as autoridades do IBM WebSphere MQ.

Visão geral da segurança

Esta coleção de tópicos apresenta os conceitos de segurança do IBM WebSphere MQ.

Os conceitos e mecanismos de segurança, conforme são aplicados a qualquer sistema de computador, são apresentados primeiro, seguido por uma discussão desses mecanismos de segurança, à medida que são implementados em IBM WebSphere MQ.

Conceitos e Mecanismos de Segurança

Esta coleção de tópicos descreve aspecto de segurança para considerar em sua instalação do IBM WebSphere MQ.

Os aspectos comumente aceitos de segurança são os seguintes:

- “Identificação e autenticação” na página 5
- “Authorization” na página 6
- “Auditing” na página 6
- “Sigilosidade” na página 7
- “Integridade de dados” na página 7

Mecanismos de Segurança são ferramentas técnicas e métodos que são utilizados para implementar serviços de segurança. Um mecanismo pode operar por conta própria, ou com outros, para fornecer um serviço específico. Exemplos de mecanismos de segurança comuns são os seguintes:

- “Criptografia” na página 7
- “Trechos de mensagens e assinaturas digitais” na página 9
- “Certificados Digitais” na página 9
- “Infraestrutura da Chave Pública (PKI)” na página 14

Ao planejar uma implementação do IBM WebSphere MQ , considere quais mecanismos de segurança você precisa para implementar esses aspectos de segurança que são importantes para você. Para obter informações sobre o que considerar depois de ler esses tópicos, consulte “Planejando para seus requisitos de segurança” na página 47.

Conceitos relacionados

“Conectando dois gerenciadores de filas usando SSL ou TLS” na página 210
As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

“Trabalhando com SSL ou TLS” na página 113
Esses tópicos fornecem instruções para executar tarefas únicas relacionadas ao uso de SSL ou TLS com o IBM WebSphere MQ.

Identificação e autenticação

Identificação é a capacidade de identificar exclusivamente um usuário de um sistema ou um aplicativo que esteja sendo executado no sistema. *Autenticação* é a capacidade de provar que um usuário ou um aplicativo é realmente quem essa pessoa ou o que esse aplicativo diz ser.

Por exemplo, considere um usuário que entre no sistema com um ID de usuário e uma senha. O sistema usa o ID de usuário para identificar o usuário. O sistema autentica o usuário no momento do logon verificando se a senha fornecida está correta.

Não repúdio

O serviço *não-repudiação* pode ser visto como uma extensão dos serviços de identificação e autenticação. Em geral, a não-repudiação é aplicada quando os dados são transmitidos eletronicamente; por exemplo, uma ordem ao corretor da bolsa para comprar ou vender ações, ou uma ordem a um banco para transferir fundos de uma conta a outra.

O objetivo geral do serviço de irrecusabilidade é poder provar que uma mensagem específica está associada a uma pessoa específica.

O serviço de não-repudiação pode conter mais de um componente, em que cada componente fornece uma função diferente. Caso o emitente da mensagem alguma vez se negue a enviá-la, o serviço de não-repudiação com a *prova de origem* pode fornecer ao receptor com completa certeza de que a mensagem foi enviada por aquele indivíduo específico. Caso o receptor da mensagem alguma vez se negue a enviá-la, o serviço de não-repudiação com a *prova de entrega* pode fornecer ao emitente com completa certeza de que a mensagem foi recebida por aquele indivíduo específico.

Na prática, provar com completa ou quase 100% de certeza, é uma tarefa difícil. No mundo real, nada é totalmente seguro. Gerenciamento de segurança está mais concentrado em gerenciar riscos a um nível que seja aceitável para os negócios. Em tal ambiente, uma expectativa mais realística do serviço de não-repudiação é ser capaz de fornecer provas que sejam admissíveis e que apóiem o seu caso em um tribunal de lei.

O não repúdio é um serviço de segurança relevante em um ambiente do IBM WebSphere MQ porque o IBM WebSphere MQ é um meio de transmitir dados eletronicamente. Por exemplo, você pode exigir provas contemporâneas de que uma mensagem específica foi enviada ou recebida por um aplicativo associado a um indivíduo em particular.

O IBM WebSphere MQ com o IBM WebSphere MQ Advanced Message Security não fornece um serviço de não repúdio como parte de sua função base. No entanto, esta documentação do produto contém sugestões sobre como você pode fornecer seu próprio serviço de não repúdio em um ambiente do WebSphere MQ gravando seus próprios programas de saída.

Conceitos relacionados

“Identificação e autenticação no IBM WebSphere MQ” na página 21
No IBM WebSphere MQ, é possível implementar a identificação e autenticação usando informações de contexto da mensagem e autenticação mútua.

Authorization

A *autorização* protege os recursos críticos em um sistema, limitando o acesso somente a usuários autorizados e seus aplicativos. Ele previne o uso não autorizado de um recurso ou o uso de um recurso de maneira não autorizada.

Conceitos relacionados

“Autorização em IBM WebSphere MQ .” na página 21
É possível usar a autorização para limitar o que indivíduos ou aplicativos específicos podem fazer em seu ambiente do IBM WebSphere MQ .

Auditing

Auditoria é o processo de gravação e verificação de eventos para detectar se qualquer atividade inesperada ou desautorizada ocorreu, ou se nenhuma tentativa foi feita para executar essa atividade.

Para obter mais informações sobre como configurar a autorização, consulte “Planejando a autorização” na página 50 e os subtópicos associados.

Conceitos relacionados

“Auditoria em IBM WebSphere MQ” na página 22

O IBM WebSphere MQ pode emitir mensagens de eventos para registrar que uma atividade incomum ocorreu.

Sigilosidade

O serviço de *confidencialidade* protege informações sensitivas de exposições não autorizadas.

Quando dados sensíveis são armazenados localmente, os mecanismos de controle de acesso podem ser suficientes para protegê-lo na hipótese de não ser possível ler os dados caso não possam ser acessados. Caso um nível mais alto de segurança seja necessário, os dados podem ser critografados.

Criptografe dados sensíveis quando são transmitidos em uma rede de comunicações, especialmente em uma rede tão insegura quanto a Internet. Em um ambiente de rede, os mecanismos de controle de acesso não são efetivos contra tentativas de intercepção de dados, como grampeamento de linha.

Integridade de dados

O serviços *integridade dos dados* detecta se houve modificação não-autorizada dos dados.

Há duas maneiras nas quais os dados podem ser alterados: acidentalmente, por meio de erros de hardware ou transmissão ou em decorrência de um ataque deliberado. Muitos produtos de hardware e protocolos de transmissão possuem mecanismos para detectar e corrigir erros de hardware e de transmissão. O propósito de serviços de integridade de dados é detectar um ataque deliberado.

O serviço de integridade dos dados tem como objetivo somente detectar se algum dado foi modificado. Não tem a meta de restaurar dados ao seu estado original caso tenha sido modificado.

Mecanismos de controle de acesso podem contribuir para a integridade dos dados pois estes não podem ser modificados caso o acesso tenha sido negado. No entanto, como na confidencialidade, os mecanismos de controle de acesso não são efetivos em ambientes de rede.

Conceitos criptográficos

Esta coleção de tópicos descreve os conceitos de criptografia aplicáveis ao WebSphere MQ

O termo *entidade* é usado para referir-se a um gerenciador de filas, um cliente MQI do WebSphere MQ , um usuário individual ou qualquer outro sistema capaz de trocar mensagens

Conceitos relacionados

“Criptografia em IBM WebSphere MQ” na página 23
O IBM WebSphere MQ fornece criptografia usando os protocolos Secure Sockets Layer (SSL) e Transport Security Layer (TSL).

Criptografia

Criptografia é o processo de conversão entre texto legível, chamado *texto simples*, e uma forma ilegível, chamada *texto cifrado*.

Isso ocorre conforme a seguir:

1. O emissor converte a mensagem de texto corrido para texto cifrado. Essa parte do processo é chamada *criptografia* (às vezes *encriptação*).
2. O texto cifrado é transmitido ao receptor.
3. O receptor converte a mensagem de texto cifrado de volta à sua forma de texto corrido. Esta parte do processo é chamada *decriptografia* (às vezes *decifração*).

Consulte o Glossário para obter uma definição de criptografia

A conversação envolve uma seqüência de operações matemáticas que alteram a aparência da mensagem durante a transmissão, mas não afetam o conteúdo. As técnicas criptográficas podem assegurar confidencialidade e proteger mensagens contra visualização não autorizada (escuta), porque uma mensagem criptografada não é compreensível. Assinaturas digitais, que fornecem uma garantia da integridade da mensagem, usam técnicas de criptografia. Consulte a “Assinaturas digitais em SSL e TLS” na página 19 para obter mais informações.

Técnicas de criptografia envolvem um algoritmo geral, tornado específico pelo uso das chaves. Há duas classes de algoritmo:

- Aqueles que exigem que ambas as partes utilizem a mesma chave. Algoritmos que utilizam uma chave compartilhada são conhecidos como algoritmos *simétricos*.Figura 1 na página 8 ilustra criptografia de chave simétrica.
- Aqueles que utilizam uma chave para criptografia e uma chave diferente para decriptografia. Um destes deve ser mantido secreto mas o outro pode ser público. Algoritmos que utilizam pares de chaves públicas e privadas são conhecidos como algoritmos *assimétricos*.Figura 2 na página 8 ilustra a criptografia de chave assimétrica, que também é conhecida como *criptografia de chave pública*.

Os algoritmos de criptografia e decriptografia utilizados podem ser públicos mas a chave secreta compartilhada e a chave privada devem ser mantidas secretas.

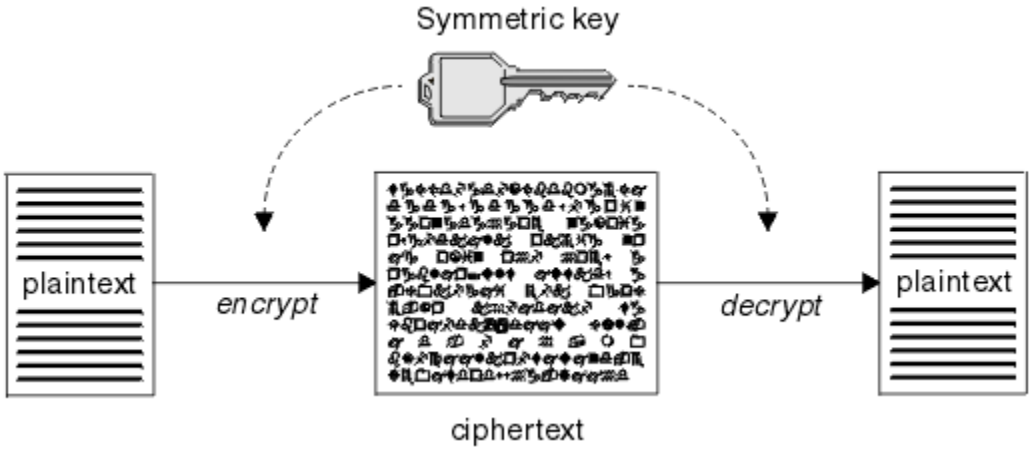


Figura 1. Criptografia de chave simétrica

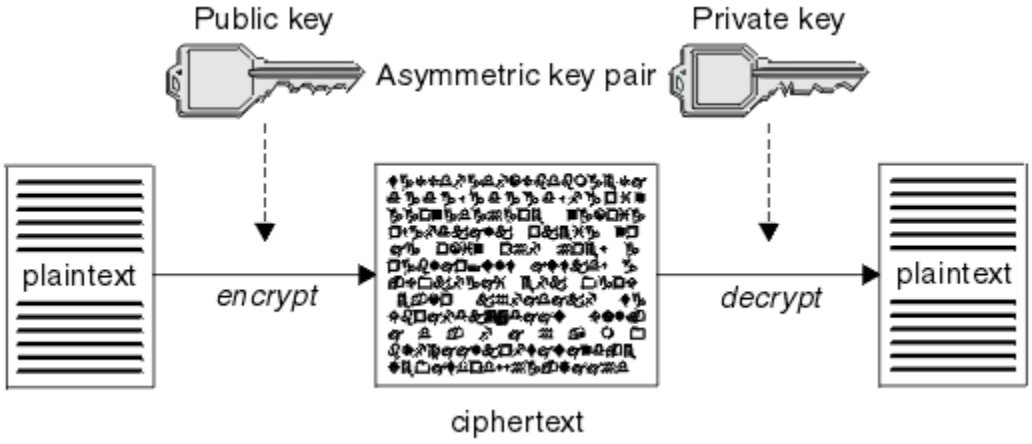


Figura 2. Criptografia de chave assimétrica

Figura 2 na página 8 mostra texto corrido criptografado com a chave pública do receptor e decriptografado com a chave privada do receptor. Somente o receptor pretendido tem a chave privada para decriptografar o texto cifrado. Observe que o emissor pode também criptografar mensagens com uma chave privada, que permite que qualquer um que tenha a chave pública do emissor decriptografe a mensagem, com a garantia de que a mensagem deve ter vindo do emissor.

Com algoritmos assimétricos, as mensagens são criptografadas com a chave pública ou privada mas podem ser decriptografados somente com a outra chave. Somente a chave privada é secreta, a chave pública pode ser conhecida por qualquer um. Com algoritmos simétricos, a chave compartilhada deve ser conhecida somente pelas duas partes. Isso é chamado de *problema de distribuição de chave* Algoritmos assimétricos são mais lentos mas têm a vantagem de que não há problema de distribuição de chave.

Outra terminologia associada com a criptografia é:

Ponto Forte
A força da criptografia é determinada pelo tamanho da chave. Algoritmos assimétricos exigem chaves grandes, por exemplo:

1024 bits	Chave assimétrica de força baixa
2048 bits	Chave assimétrica de força média
4096 bits	Chave assimétrica de força alta

As chaves simétricas são menores: as chaves de 256 bits fornecem criptografia avançada.

Algoritmo de cifra de bloco
Estes algoritmos criptografam dados por blocos. Por exemplo, o algoritmo RC2 do RSA Data Security Inc. usa blocos de 8 bytes de comprimento. Algoritmos de bloco são geralmente mais lentos do que algoritmos de fluxo.

Algoritmo de cifra de fluxo
Estes algoritmos operam em cada byte de dados. Algoritmos de fluxo são geralmente mais rápidos do que algoritmos de bloco.

Trechos de mensagens e assinaturas digitais

Uma compilação de mensagens é uma representação numérica de tamanho fixo do conteúdo de uma mensagem, calculada por uma função hash. Uma compilação de mensagem pode ser criptografada, formando uma assinatura digital.

As mensagens são inerentemente variáveis em tamanho. O trecho da mensagem é uma representação numérica de tamanho fixo do conteúdo de uma mensagem. Uma compilação de mensagem é calculada por uma função hash, que é uma transformação que atende a dois critérios:

- A função hash deve ser unidirecional. Não deve ser possível reverter a função para encontrar a mensagem correspondente a um trecho de mensagem específico, a não ser testando todas as mensagens possíveis.
- Deve ser computacionalmente impossível encontrar duas mensagens que executem hash para a mesma compilação.

A compilação de mensagens é enviada junto com a própria mensagem. O destinatário pode gerar uma compilação para a mensagem e compará-la com a compilação do emissor. A integridade da mensagem é verificada quando os dois trechos da mensagem são os mesmos. Qualquer violação da mensagem durante a transmissão quase certamente resultará em uma compilação de mensagem diferente.

Um trecho da mensagem criado usando uma chave simétrica secreta é conhecido como um código de autenticação de mensagem (MAC), pois ele pode fornecer garantia de que a mensagem não foi modificada.

O emissor pode também gerar um trecho da mensagem e, em seguida, criptografar a compilação usando a chave privada de um par de chaves assimétricas, formando uma assinatura digital. A assinatura deve, então, ser decriptografada pelo receptor, antes de compará-la com uma compilação gerada localmente.

Conceitos relacionados
“Assinaturas digitais em SSL e TLS” na página 19
Uma assinatura digital é formada pela criptografia de uma representação de uma mensagem. A criptografia utiliza a chave privada do assinante e, para eficiência, geralmente opera em uma compilação de mensagens, ao invés de na mensagem em si.

Certificados Digitais

Os certificados digitais são protegidos contra personificação, certificando que uma chave pública pertence a uma entidade especificada. Eles são emitidos por uma autoridade de certificação.

Certificados digitais fornecem proteção contra identidades falsas, porque um certificado digital liga uma chave pública a seu proprietário, seja este um indivíduo, um gerenciador de filas ou alguma outra entidade. Certificados digitais também são conhecidos como certificados de chave pública, porque eles

oferecem garantias sobre a propriedade de uma chave pública quando você utiliza um esquema de chave assimétrica. Um certificado digital contém a chave pública de uma entidade e é uma confirmação de que a chave pública pertence àquela entidade:

- Quando o certificado for para uma entidade individual, ele é chamado de *certificado pessoal* ou *certificado de usuário*.
- Quando o certificado for para uma Autoridade de Certificação, ele é chamado de *certificado de autoridade de certificação* ou *certificado de assinante*.

Se chaves públicas forem enviadas diretamente por seu proprietário a outra entidade, há um risco de que a mensagem possa ser interceptada e a chave pública ser substituída por outra. Isso é conhecido como *ataque humano intermediário* . A solução para este problema é trocar chaves públicas por meio de terceiros confiáveis, o que proporcionará ao usuário uma garantia segura de que a chave pública realmente pertence à entidade com a qual você está se comunicando. Em vez de enviar sua chave pública diretamente, peça aos terceiros confiáveis para incorporá-la a um certificado digital. Os terceiros confiáveis que emitem certificados digitais são chamados de Autoridade de Certificação (CA), conforme descrito em “Autoridades de Certificação” na página 11.

O Que é um Certificado Digital
Os certificados digitais contêm partes específicas de informações, conforme determinado pelo padrão de X.509.

Os certificados digitais usados pelo WebSphere MQ estão em conformidade com o padrão X.509 , que especifica as informações necessárias e o formato para enviá-los. X.509 é a estrutura de Autenticação que faz parte da série X.500 de padrões.

Os certificados digitais contêm, no mínimo, as seguintes informações sobre a entidade que está sendo certificada:

- A chave pública do proprietário
- O Nome Distinto do proprietário
- O Nome Distinto da CA que emitiu o certificado
- A data a partir da qual o certificado é válido
- A data de vencimento do certificado
- O número da versão do formato de dados do certificado conforme definido no X.509. A versão atual do padrão X.509 é Versão 3, e a maioria dos certificados está em conformidade com essa versão.
- Um número de série. Esse é um identificador exclusivo designado pelo CA que emitiu o certificado. O número de série é exclusivo dentro do CA que emitiu o certificado: não há dois certificados assinados pelo certificado de CA que têm o mesmo número de série.

Um certificado X.509 Versão 2 também contém um Identificador do Emissor e um Identificador de Assunto, e um certificado X.509 Versão 3 pode conter várias extensões. Algumas extensões de certificado, como a extensão de Restrição Básica, são *padrão*, mas outras são específicas à implementação. Uma extensão pode ser *crítica*, no caso em que um sistema deve ser capaz de reconhecer o campo. Se ele não reconhecer o campo, deverá rejeitar o certificado. Se uma extensão não for crítica, o sistema poderá ignorá-la, se não a reconhecer.

A assinatura digital em um certificado pessoal é gerada usando a chave privada do CA que assinou esse certificado. Qualquer pessoa que precisa verificar o certificado pessoal pode usar a chave pública do CA para fazer isso. O certificado do CA contém sua chave pública.

Os certificados digitais não contêm sua chave privada. Você deve manter sua chave privada em segredo.

Requisitos para certificados pessoais
O WebSphere MQ suporta certificados digitais que estão em conformidade com o padrão X.509 Ele requer a opção de autenticação de cliente.

Como o IBM WebSphere MQ é um sistema ponto a ponto, ele é visualizado como autenticação de cliente na terminologia do SSL. Portanto, qualquer certificado pessoal usado para autenticação SSL precisa permitir uma utilização chave de autenticação de cliente. Nem todos os certificados de servidor têm esta

opção ativada, de forma que o fornecedor do certificado pode precisar ativar a autenticação de cliente no AC raiz do certificado seguro.

Além dos padrões que especificam o formato de dados para um certificado digital, há também os padrões para determinar se um certificado é válido. Essas normas foram atualizadas com o passar do tempo, a fim de evitar determinados tipos de violação de segurança. Por exemplo, os certificados mais antigos do X.509 versão 1 e 2 não indicam se o certificado pode ser legitimamente usado para assinar outros certificados. Era possível, portanto, para um usuário mal intencionado, obter um certificado pessoal de uma fonte legítima e criar novos certificados projetados para personificar outros usuários.

Ao usar certificados X.509 versão 3, o BasicConstraints e as extensões de certificado KeyUsage são usados para especificar quais certificados podem assinar legitimamente outros certificados. O padrão IETF RFC 5280 especifica uma série de regras de validação de certificado que o software de aplicativo compatível deve implementar para impedir ataques de personificação. Um conjunto de regras de certificado é conhecido como uma política de validação de certificado.

Para obter mais informações sobre as políticas de validação de certificado no IBM WebSphere MQ, consulte “Políticas de validação de certificado no IBM WebSphere MQ” na página 34.

Autoridades de Certificação
Uma autoridade de certificação (CA) é um terceiro confiável que emite certificados digitais para fornecer a garantia de que a chave pública de uma entidade verdadeiramente pertença àquela entidade.

As funções de uma CA são:

- Ao receber um pedido de um certificado digital, verifique a identidade do solicitante antes de construir, assinar e devolver o certificado pessoal
- Fornecer a chave pública da própria CA em seu certificado de CA
- Publicar listas de certificados que não são mais confiáveis em uma CRL (Lista de Revogação de Certificados). Para obter informações adicionais, consulte “Trabalhando com Certificados Revogados” na página 153
- Para fornecer acesso ao status de revogação de certificado, operando um respondente servidor de respondente do OCSP

Nomes Distintos
O DN (Distinguished Name) identifica de modo exclusivo uma entidade em um certificado X.509.

Os seguintes tipos de atributo são comumente encontrados no DN:

SERIALNUMBER	Número de série do certificado
MAIL	Endereço eletrônico
E	Endereço de e-mail (descontinuado na preferência para MAIL)
UID ou USERID	Identificador de usuário
CN	Nome Comum
T	Título
OU	Nome de Unidade Organizacional
DC	Componente de domínio
O	Nome da organização
STREET	Rua / Primeira linha do endereço
L	Nome da localidade
ST (ou SP ou S)	Nome do estado ou região
PC	Código Postal / Código de Endereçamento Postal

C	País
UNSTRUCTUREDNAME	Nome do host
UNSTRUCTUREDADDRESS	Endereço IP
DNQ	Qualificador de Nome Distinto

O padrão X.509 define outros atributos que normalmente não fazem parte do DN, mas podem fornecer extensões opcionais para o certificado digital.

O padrão X.509 faz com que um DN seja especificado em formato de cadeia. Por exemplo:

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

O Nome Comum (CN) pode descrever um usuário individual ou qualquer outra entidade, por exemplo, um servidor da Web.

O DN pode conter diversos atributos OU e DC. Apenas uma instância de cada um dos outros atributos é permitida. A ordem das entradas de OU é significante: ela especifica uma hierarquia de nomes de Unidades Organizacionais, com a unidade de mais alto nível primeiro. A ordem das entradas DC também é significativa.

IBM WebSphere MQ tolera certos DNs malformados. Para obter mais informações, consulte WebSphere MQ para valores SSLPEER.

Conceitos relacionados

"O Que é um Certificado Digital" na página 10
Os certificados digitais contêm partes específicas de informações, conforme determinado pelo padrão de X.509.

Obtendo certificados pessoais a partir de uma autoridade de certificação
É possível obter um certificado a partir de uma autoridade de certificação (CA) externa confiável.

Você obtém um certificado digital enviando informações a um CA na forma de uma solicitação de certificado. O padrão X.509 define um formato para estas informações, mas alguns CAs têm seu próprio formato. Solicitações de certificado geralmente são geradas pela ferramenta de gerenciamento de certificado que seu sistema usa, por exemplo, a ferramenta iKeyman nos sistemas UNIX, Linux®e Windows e RACF no z/OS. As informações contêm seu Nome Distinto e sua chave pública. Quando sua ferramenta de gerenciamento de certificados gera seu pedido de certificado, também gera sua chave privada, que você deve manter segura. Nunca distribua sua chave privada.

Quando a CA recebe seu pedido, a autoridade verifica sua identidade antes de construir o certificado e devolvê-lo a você como um certificado pessoal.

Figura 3 na página 12 ilustra o processo de obtenção de um certificado digital de uma CA.

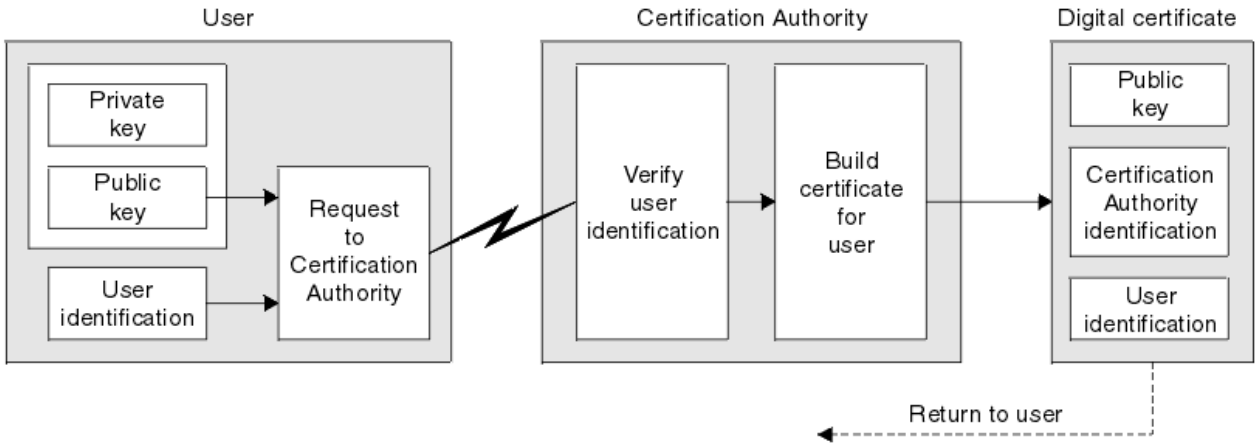


Figura 3. Obtendo um certificado digital

No diagrama:

- A "identificação de usuário" inclui o Nome Distinto do Assunto.
- A "identificação da Autoridade de Certificação" inclui o Nome Distinto da CA que está emitindo o certificado.
-

Os certificados digitais contêm campos adicionais diferentes dos mostrados no diagrama. Para obter mais informações sobre os outros campos em um certificado digital, consulte "O Que é um Certificado Digital" na página 10.

Como Funcionam as Cadeias de Certificados
Quando você receber o certificado de outra entidade, você pode precisar utilizar uma *cadeia de certificados* para obter o certificado *CA raiz*.

A cadeia de certificados, também conhecida como o *caminho de certificação* , é uma lista de certificados usados para autenticar uma entidade A cadeia, ou caminho, começa com o certificado daquela entidade, e cada certificado na cadeia é assinado pela entidade identificada pelo próximo certificado na cadeia. A cadeia termina com um certificado de CA raiz. O certificado de autoridade de certificação raiz é sempre assinado pela própria autoridade de certificação (CA). As assinaturas de todos os certificados na cadeia devem ser verificadas até que o certificado de CA raiz seja alcançado.

Figura 4 na página 13 ilustra um caminho de certificação do proprietário do certificado para a CA raiz, onde a cadeia de confiança começa.

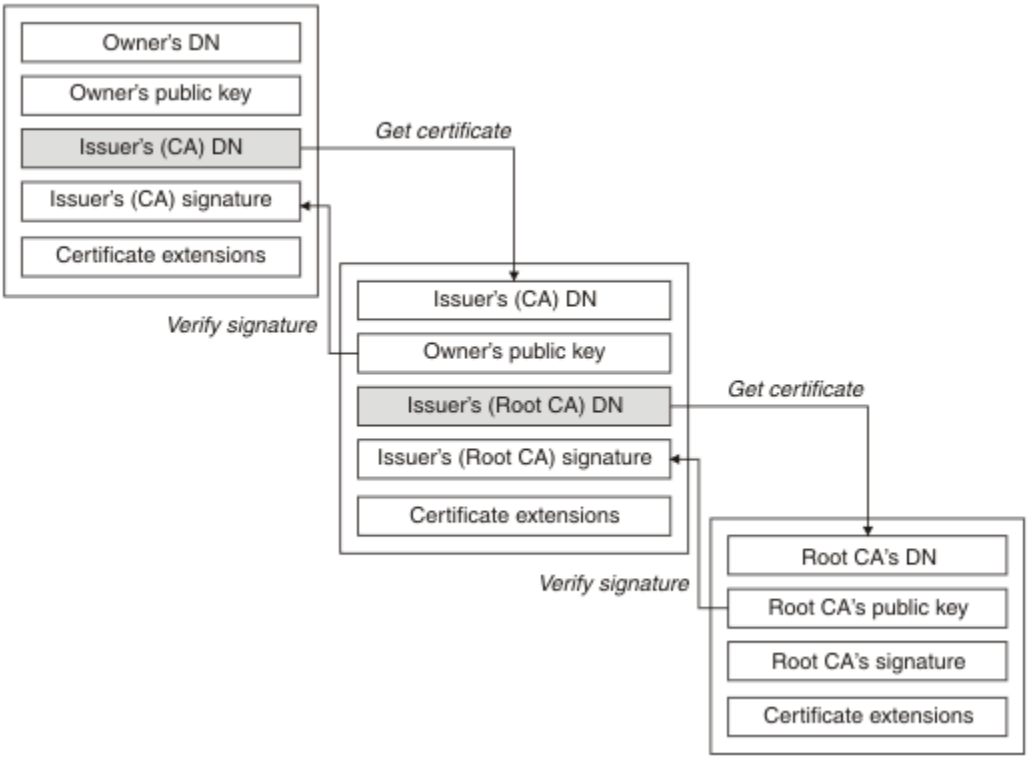


Figura 4. Cadeia de confiança

Cada certificado pode conter uma ou mais extensões. Um certificado pertencente a uma CA geralmente contém uma extensão BasicConstraints com o sinalizador isCA configurado para indicar que é permitido que ele assine outros certificados.

Quando os Certificados Não São Mais Válidos
Os certificados digitais podem vencer ou serem revogados.

Certificados digitais também são emitidos por um período fixo e não são válidos depois de suas datas de expiração.

Consulte o Glossário para obter uma definição de expiração de certificado

Certificados podem ser revogados por várias razões, incluindo:

- O proprietário mudou para uma organização diferente.
- A chave privada não é mais secreta.

WebSphere MQ pode verificar se um certificado foi revogado enviando uma solicitação para um respondente Online Certificate Status Protocol (OCSP) (somente nos sistemas UNIX, Linux e Windows). Como alternativa, eles podem acessar uma CRL em um servidor LDAP A revogação do OCSP e as informações da CRL são publicadas por uma autoridade de certificação. Para obter informações adicionais, consulte "Trabalhando com Certificados Revogados" na página 153.

Infraestrutura da Chave Pública (PKI)

O PKI (Public Key Infrastructure) é um sistema de recursos, políticas, e serviços que suportam a utilização de criptografia de tecla pública para autenticar as partes envolvidas na transação.

Não há nem um único padrão que define os componentes de uma infraestrutura de chave pública, mas uma infraestrutura de chave pública geralmente inclui autoridades de certificação (CAs) e Autoridades de registro (RAs). Os CAs fornecem os serviços a seguir:

- Emissão de certificados digitais
- Validação de certificados digitais
- Revogação de certificados digitais
- Distribuição de teclas públicas

Os padrões X.509 fornecem a base para a infraestrutura de chave pública padrão de mercado.

Consulte "Certificados Digitais" na página 9 para obter mais informações sobre certificados digitais e autoridades de certificação (CAs). RAs verificam se as informações fornecidas quando certificados digitais são solicitados. Se o RA verificar a informação, o CA pode emitir um certificados digital ao solicitante.

Um PKI também pode fornecer as ferramentas para o gerenciamento de certificados digitais ou teclas públicas. Um PKI, às vezes, é descrito como uma *hierarquia de confiança* para o gerenciamento de certificados digitais, mas a maioria das definições incluem serviços adicionais. Algumas definições incluem serviços de criptografia e de assinatura digital, mas não são essenciais para a operação de um PKI.

Protocolos de Segurança Criptográficos: SSL e TLS

Os protocolos de criptografia fornecem conexões seguras, permitindo que dois grupos de comuniquem com privacidade e integridade de dados. O protocolo Transport Layer Security (TLS) surgiu desse do Secure Sockets Layer (SSL). IBM WebSphere MQ suporta SSL e TLS.

Os objetivos principais de ambos os protocolos são fornecer confidencialidade (às vezes referida como *privacidade*), integridade de dados, identificação e autenticação usando certificados digitais.

Embora os dois protocolos sejam semelhantes, as diferenças são suficientemente significantes que o SSL 3.0 e as várias outras versões do TLS não interoperam.

Conceitos relacionados

"Protocolos de segurança no IBM WebSphere MQ" na página 23
O IBM WebSphere MQ suporta os protocolos Transport Layer Security (TLS) e Secure Sockets Layer (SSL) para fornecer segurança em nível de link para canais de mensagens e canais de MQI.

Conceitos do Secure Sockets Layer (SSL) e Transport Layer Security (TLS)

Os protocolos SSL e TLS permitem que duas partes se identifiquem, se autentiquem e se comuniquem com confidencialidade e integridade de dados. O protocolo TLS surgiu do protocolo Netscape SSL 3.0, mas o TLS e o SSL não interoperam.

Os protocolos SSL e TLS fornecem segurança de comunicações sobre a Internet e permitem que os aplicativos do cliente/servidor se comuniquem de uma maneira confidencial e confiável. Os protocolo possuem duas camadas: um Protocolo de Registro e um Protocolo de Handshake e, eles são dispostos em camadas sobre um protocolo de transporte como TCP/IP. Ambos usam técnicas de criptografia assimétrica e simétrica.

Uma conexão SSL ou TLS é iniciada por um aplicativo, que se torna o cliente SSL ou TLS. O aplicativo que recebe a conexão se torna o servidor SSL ou TLS. Cada nova sessão é iniciada com um handshake, conforme definido pelos protocolos SSL ou TLS.

Uma lista integral de CipherSpecs suportados por IBM WebSphere MQ é fornecida em “Especificando CipherSpecs” na página 221.

Para obter mais informações sobre o protocolo SSL, consulte as informações fornecidas em https://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt. Para obter mais informações sobre o protocolo TLS, consulte as informações fornecidas pelo Grupo de trabalho de TLS no website do Internet Engineering Task Force em https://www.ietf.org

Uma visão geral do handshake SSL ou TLS

O handshake SSL ou TLS permite que o cliente e o servidor SSL ou TLS estabeleça as chaves secretas com as quais eles se comunicam.

Esta seção fornece um resumo das etapas que permitem que o cliente e o servidor SSL ou TLS se comuniquem entre si.

- Aceitar a versão do protocolo a ser usada.
- Selecionar algoritmos criptográficos, que são descritos em .
- Autenticar um ao outro com a troca e validação de certificados digitais.
- Utilizar técnicas de criptografia assimétrica para gerar uma chave compartilhada secreta, que evita o problema de distribuição de chaves. SSL ou TLS, em seguida, use a chave compartilhada para a criptografia simétrica de mensagens, que é mais rápida que a criptografia assimétrica.

Para obter mais informações sobre a importação de certificados, consulte a seção relevante para sua plataforma em .

Numa visão geral, as etapas envolvidas no protocolo de reconhecimento de SSL são as seguintes:

1. O cliente SSL ou TLS envia uma mensagem "client hello" que lista informações criptográficas, como a versão de SSL ou TLS e, na ordem de preferência do cliente, os CipherSuites suportados pelo cliente. A mensagem também contém uma cadeia de bytes aleatória que é utilizada em cálculos subsequentes. O protocolo permite que o "client hello" inclua métodos de compactação de dados suportados pelo cliente.
2. O servidor SSL ou TLS responde com uma mensagem "server hello" que contém o CipherSuite escolhido pelo servidor a partir da lista fornecida pelo cliente, o ID de sessão e outra sequência de bytes aleatória. O servidor também envia seu certificado digital. Se o servidor exigir um certificado digital para a autenticação do cliente, o servidor envia um "pedido de certificado de cliente" que inclui uma lista dos tipos de certificados suportados e os Nomes Distintos de Autoridades de Certificação (CAs) aceitáveis.
3. O cliente SSL ou TLS verifica o certificado digital do servidor. Para obter informações adicionais, consulte “Como SSL e TLS Fornecem Identificação, Autenticação, Confidencialidade e Integridade” na página 16.
4. O cliente SSL ou TLS envia a sequência de bytes aleatória que permite ao cliente e ao servidor calcular a chave secreta a ser usada para a criptografia de dados de mensagens subsequentes. A própria cadeia de bytes aleatória é criptografada com a chave pública do servidor.
5. Se o servidor SSL ou TLS enviou uma "solicitação de certificado de cliente", o cliente envia uma sequência de bytes aleatória criptografada com a chave privada do cliente, juntamente com o certificado digital do cliente ou um "nenhum alerta de certificado digital". Este alerta é apenas um aviso, mas com algumas implementações, o protocolo de reconhecimento falha em caso de obrigatoriedade da autenticação do cliente.

6. O servidor SSL ou TLS verifica o certificado do cliente. Para obter informações adicionais, consulte “Como SSL e TLS Fornecem Identificação, Autenticação, Confidencialidade e Integridade” na página 16.
7. O cliente SSL ou TLS envia a um servidor uma mensagem de "concluído", que é criptografada com a chave secreta, indicando que a parte do handshake do cliente está concluída.
8. O servidor SSL ou TLS envia ao cliente uma mensagem "concluído", que é criptografada com a chave secreta, indicando que a parte do handshake do servidor está concluída.
9. Para a duração da sessão de SSL ou TLS, o servidor e o cliente agora podem trocar mensagens que são criptografadas simetricamente com a chave secreta compartilhada.

Figura 5 na página 16 ilustra o handshake do SSL ou TLS.

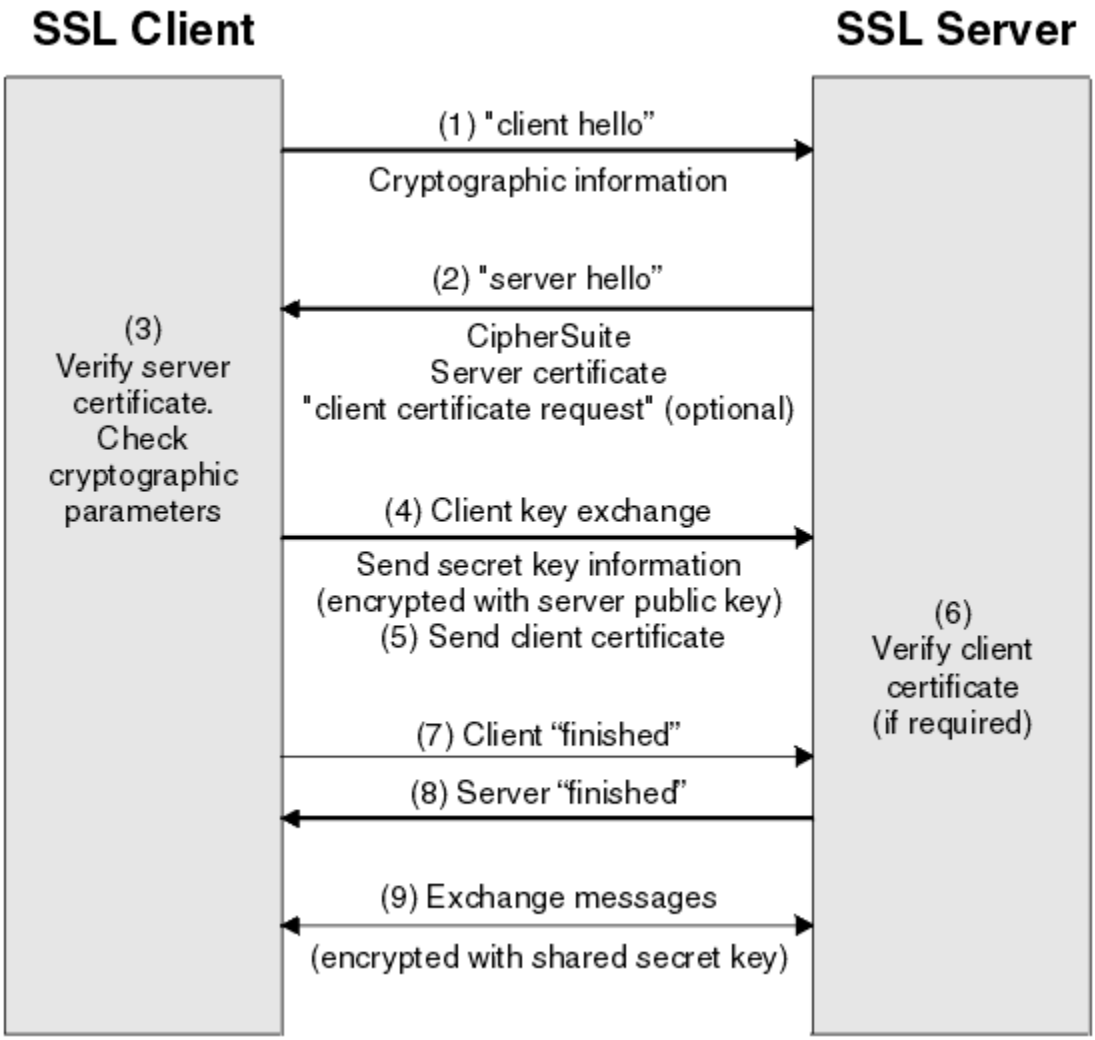


Figura 5. Visão geral do handshake do SSL ou TLS

Como SSL e TLS Fornecem Identificação, Autenticação, Confidencialidade e Integridade

Durante a autenticação do cliente e do servidor, existe uma etapa que requer que os dados sejam criptografados com uma das chaves de um par de chaves assimétricas e decriptografados com a outra chave do par. Um trecho da mensagem é usado para fornecer integridade.

Para obter uma visão geral das etapas envolvidas no handshake TLS, consulte “Uma visão geral do handshake SSL ou TLS” na página 15..

Como SSL e TLS Fornecem Autenticação

Para autenticação do servidor, o cliente utiliza a chave pública do servidor para criptografar os dados que são utilizados para calcular a chave secreta. O servidor poderá gerar a chave secreta somente se puder descriptografar esses dados com a chave privada correta.

Para a autenticação do cliente, o servidor utiliza a chave pública do certificado do cliente para decriptografar os dados enviados pelo cliente durante a etapa “5” na página 15 do protocolo de reconhecimento. A troca de mensagens concluídas que são criptografadas com a chave secreta (etapas “7” na página 16 e “8” na página 16 na visão geral) confirma que a autenticação está completa.

Se alguma das etapas de autenticação falhar, o protocolo de reconhecimento falhará e a sessão será encerrada.

A troca de certificados digitais durante o handshake do SSL ou TLS faz parte do processo de autenticação. Para obter informações adicionais sobre como os certificados fornecem proteção contra personificação, consulte as informações relacionadas. Os certificados requeridos são os seguintes, em que CA X emite o certificado para o cliente SSL ou TLS e CA Y emite o certificado para o servidor SSL ou TLS:

Apenas para autenticação de servidor, o servidor SSL ou TLS precisa:

- O certificado pessoal emitido para o servidor pela autoridade de certificação Y
- Da chave privada do servidor

e o cliente SSL ou TLS precisa:

- O certificado de CA para CA Y

Se o servidor SSL ou TLS requerer autenticação de cliente, o servidor verificará a identidade do cliente verificando o certificado digital do cliente com a chave pública para a CA que emitiu o certificado pessoal para o cliente, neste caso, CA X . Para autenticação de servidor e cliente, o servidor precisa:

- O certificado pessoal emitido para o servidor pela autoridade de certificação Y
- Da chave privada do servidor
- O certificado de CA para CA X

e o cliente precisa:

- O certificado pessoal emitido para o cliente pela autoridade de certificação X
- Da chave privada do cliente
- O certificado de CA para CA Y

O cliente e o servidor SSL ou TLS podem precisar de outros certificados da CA para formarem uma cadeia de certificados para o certificado da CA raiz. Para obter informações adicionais sobre as cadeias de certificados, consulte as informações relacionadas.

O que Acontece Durante a Verificação de Certificado

Conforme observado nas etapas “3” na página 15 e “6” na página 16 da visão geral, o cliente SSL ou TLS verifica o certificado do servidor e o servidor SSL ou TLS verifica o certificado do cliente. Existem quatro aspectos para esta verificação:

1. A assinatura digital é verificada (consulte “Assinaturas digitais em SSL e TLS” na página 19).
2. A cadeia de certificados é verificada; é necessário ter certificados de autoridade de certificação intermediária (consulte “Como Funcionam as Cadeias de Certificados” na página 13).
3. As datas de expiração e de ativação e o período de validade são verificados.
4. O status de revogação do certificado é verificado (consulte “Trabalhando com Certificados Revogados” na página 153).

Reconfiguração de Chave Secreta

Durante um handshake do SSL ou TLS, uma *chave secreta* é gerada para criptografar dados entre o cliente e o servidor SSL ou TLS. A chave secreta é utilizada em uma fórmula matemática que é aplicada aos dados para transformar o texto puro em texto criptografado ilegível, e texto criptografado em texto puro.

A chave secreta é gerada a partir do texto aleatório enviado como parte da handshake, e é usada para criptografar um texto simples para texto cifrado. A chave secreta também é utilizada no algoritmo MAC (Message Authentication Code), que é utilizado para determinar se uma mensagem foi alterada. Consulte a "Trechos de mensagens e assinaturas digitais" na página 9 para obter mais informações.

Caso a chave secreta seja descoberta, o texto puro de uma mensagem poderia ser decifrado a partir do texto criptografado, ou o resumo da mensagem poderia ser calculado, permitindo que mensagens sejam alteradas sem que isso seja detectado. Mesmo para um algoritmo complexo, o texto puro pode eventualmente ser descoberto aplicando todas as transformações matematicamente possíveis ao texto criptografado. Para minimizar a quantidade de dados que podem ser decifrados ou alterados caso a chave secreta seja descoberta, a chave secreta pode ser renegociada periodicamente. Quando a chave secreta for renegociada, a chave secreta anterior não poderá mais ser usada para decriptografar dados que foram criptografados com a nova chave secreta.

Como SSL e TLS Fornecem Confidencialidade

SSL e TLS usam uma combinação de criptografia simétrica e assimétrica para assegurar a privacidade da mensagem. Durante o handshake do SSL ou TLS, o cliente e o servidor SSL ou TLS concordam em relação a um algoritmo de criptografia e uma chave secreta compartilhada que serão usados apenas para uma sessão. Todas as mensagens transmitidas entre o cliente e o servidor SSL ou TLS serão criptografadas usando esse algoritmo e essa chave, o que assegura que a mensagem continuará sendo particular se for interceptada. O SSL suporta um amplo intervalo de algoritmos criptográficos. Como SSL e TLS usam criptografia assimétrica quando transportam a chave secreta compartilhada, não há nenhum problema de distribuição de chaves. Para obter mais informações sobre técnicas de criptografia, consulte "Criptografia" na página 7.

Como SSL e TLS Fornecem Integridade

SSL e TLS fornecem integridade de dados calculando um trecho da mensagem. Para obter informações adicionais, consulte "Integridade de dados de mensagens" na página 230.

O uso de SSL ou TLS assegura integridade de dados, contanto que o CipherSpec na definição de canal use um algoritmo hash, conforme descrito na tabela em "Especificando CipherSpecs" na página 221.

Especificamente, se a integridade de dados for um problema, você deve evitar a escolha de um CipherSpec cujo algoritmo hash esteja listado como "Nenhum". O uso de MD5 também é fortemente desencorajado, pois isso agora é muito antigo e não é mais seguro para a maioria dos propósitos práticos.

CipherSpecs e CipherSuites

Os protocolos de segurança criptográficos devem concordar sobre os algoritmos usados por uma conexão segura. CipherSpecs e CipherSuites definem combinações específicas de algoritmos.

Um CipherSpec identifica uma combinação de algoritmo de criptografia e algoritmo de código de autenticação de mensagem (MAC). As extremidades de uma conexão TLS ou SSL devem concordar com o mesmo CipherSpec para serem capazes de se comunicar.

Importante: Ao lidar com canais do IBM WebSphere MQ, você usa um CipherSpec. Ao lidar com canais do Java, canais do JMS ou canais do MQTT, especifique um CipherSuite.

Para obter mais informações sobre CipherSpecs, consulte "Especificando CipherSpecs" na página 221.

Um CipherSuite é um conjunto de algoritmos criptográficos usados por uma conexão SSL ou TLS Um conjunto compreende três algoritmos distintos:

- O algoritmo de troca de chave e autenticação, usado durante o handshake
- O algoritmo de criptografia, utilizado para codificar os dados

- O algoritmo MAC (Message Authentication Code), utilizado para gerar a compilação de mensagem

Há várias opções para cada componente do conjunto, mas somente certas combinações são válidas quando especificadas para uma conexão SSL ou TLS. O nome de um CipherSuite válido define a combinação de algoritmos utilizada. Por exemplo, o CipherSuite SSL_RSA_WITH_RC4_128_MD5 especifica:

- O algoritmo de troca de chaves e autenticação RSA
- O algoritmo de criptografia RC4, usando uma chave de 128 bits
- O algoritmo MD5 MAC

Vários algoritmos estão disponíveis para troca de chaves e autenticação, mas o algoritmo RSA é atualmente o mais amplamente utilizado. Há mais variedade nos algoritmos de criptografia e algoritmos MAC que são utilizados.

Assinaturas digitais em SSL e TLS

Uma assinatura digital é formada pela criptografia de uma representação de uma mensagem. A criptografia utiliza a chave privada do assinante e, para eficiência, geralmente opera em uma compilação de mensagens, ao invés de na mensagem em si.

Assinaturas digitais variam com os dados sendo assinados, diferente de com assinaturas manuscritas, que não dependem do conteúdo do documento sendo assinado. Se duas mensagens diferentes forem assinadas digitalmente pela mesma entidade, as duas assinaturas diferirão, mas ambas poderão ser verificadas com a mesma chave pública, ou seja, a chave pública da entidade que assinou as mensagens.

As etapas do processo de assinatura digital são as seguintes:

1. O emissor calcula uma compilação de mensagens, e então a criptografa, utilizando a chave privada do emissor, formando a assinatura digital.
2. O emissor transmite a assinatura digital com a mensagem.
3. O receptor decriptografa a assinatura digital utilizando a chave pública do emissor, gerando novamente a compilação de mensagens do emissor.
4. O receptor calcula uma compilação de mensagem recebida de dados de mensagem e verifica se as duas compilações são as mesmas.

Figura 6 na página 19 ilustra este processo.

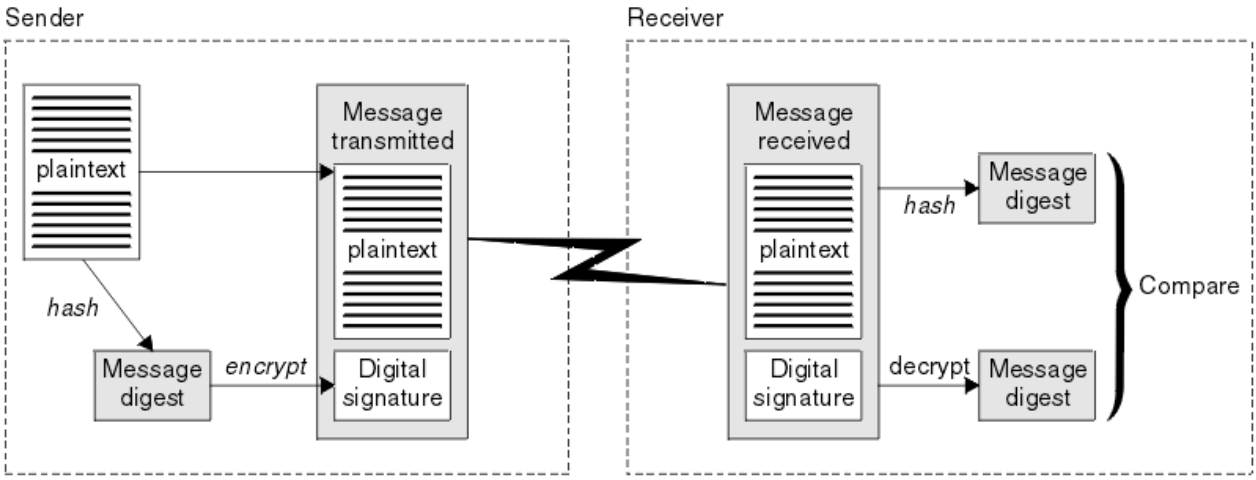


Figura 6. O processo de assinatura digital

Se a assinatura digital for verificada, o receptor saberá que:

- A mensagem não foi modificada durante a transmissão.
- A mensagem foi enviada pela entidade que declara tê-la enviado.

Assinaturas digitais são parte dos serviços de integridade e autenticação. Assinaturas digitais também proporcionam prova de origem. Somente o emissor conhece a chave privada, que fornece forte evidência de que o emissor é o originador da mensagem.

Nota: Você pode também criptografar a própria mensagem, que protegerá a confidencialidade das informações da mensagem.

Federal Information Processing Standards

O governo dos EUA produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. O National Institute for Standards and Technology (NIST) é um órgão importante ligado a sistemas e segurança de TI. O NIST produz recomendações e normas, inclusive Federal Information Processing Standards (FIPS).

Um dessas normas significativa é FIPS 140-2, que requer o uso de algoritmos criptográficos fortes. FIPS 140-2 também especifica requisitos para algoritmos hashing a serem usados para proteger pacotes contra modificação em trânsito.

O IBM WebSphere MQ fornece suporte a FIPS 140-2 quando tiver sido configurado para tal.

Ao longo do tempo, analistas desenvolvem ataques contra algoritmos de criptografia e hashing existentes. Novos algoritmos são adotados para resistir a esses ataques. FIPS 140-2 é atualizada periodicamente para considerar essas mudanças.

Criptografia do Conjunto B da Agência Nacional de Segurança (NSA)

O governo dos Estados Unidos da América produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. A Agência Nacional de Segurança dos Estados Unidos (NSA) recomenda um conjunto de algoritmos criptográficos interoperável em seu padrão do Conjunto B.

O padrão do Conjunto B especifica um modo de operação no qual somente um conjunto específico de algoritmo criptográfico seguros são usados. O padrão do Conjunto B especifica:

- O algoritmo de criptografia (AES)
- O algoritmo de troca de chave (Diffie-Hellman da curva elíptica, também conhecido como ECDH)
- O algoritmo de assinatura digital (Algoritmo de assinatura digital da curva elíptica, também conhecido como ECDSA)
- Os algoritmos hash (SHA-256 ou SHA-384)

Além disso, o padrão de IETF RFC 6460 especifica o Conjunto B compatível com perfis que definem a configuração detalhada do aplicativo e o comportamento necessário para estar em conformidade com o padrão do Conjunto B. Ele define dois perfis:

1. Um perfil compatível com o Conjunto B para uso com o TLS versão 1.2. Quando configurado para operação compatível com Suite B, apenas o conjunto restrito de algoritmos criptográficos listados acima será usado.
2. Um perfil de transição para uso com o TLS versão 1.0 ou TLS versão 1.1. Este perfil permite a interoperabilidade com servidores não compatíveis com o Conjunto B. Quando configurado para a operação transicional do Conjunto B, a criptografia adicional e os algoritmos de hash podem ser usados.

O padrão do Conjunto B é conceitualmente semelhante ao FIPS 140-2, porque restringe o conjunto de algoritmos criptográficos ativados, a fim de fornecer um nível de garantia de segurança.

Nos sistemas Windows, UNIX e Linux , o WebSphere MQpode ser configurado para estar em conformidade com o perfil TLS compatível com o Suite B 1.2 , mas não suporta o perfil de transição do Suite B. Veja informações adicionais na publicação "Criptografia do Conjunto B da NSA no IBM WebSphere MQ" na página 31.

Informações relacionadas

"Federal Information Processing Standards" na página 20
O governo dos EUA produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. O National Institute for Standards and Technology (NIST) é um órgão importante

ligado a sistemas e segurança de TI. O NIST produz recomendações e normas, inclusive Federal Information Processing Standards (FIPS).

IBM WebSphere MQ mecanismos de segurança

Esta coleção de tópicos explica como é possível implementar os vários conceitos de segurança no IBM WebSphere MQ.

O IBM WebSphere MQ fornece mecanismos para implementar todos os conceitos de segurança introduzidos no “Conceitos e Mecanismos de Segurança” na página 5 Eles são discutidos em mais detalhes nas seções a seguir.

Identificação e autenticação no IBM WebSphere MQ

No IBM WebSphere MQ, é possível implementar a identificação e autenticação usando informações de contexto da mensagem e autenticação mútua.

Aqui estão alguns exemplos da identificação e autenticação em um ambiente do IBM WebSphere MQ:

- Toda mensagem pode conter informações sobre *contexto de mensagem*. Essas informações ficam retidas no descritor de mensagens. Elas podem ser geradas pelo gerenciador de filas quando uma mensagem é colocada em uma fila por um aplicativo. Como alternativa, o aplicativo pode fornecer a informação caso o ID de usuário associado ao aplicativo tenha autorização para assim fazê-lo.

  A informação de contexto em uma mensagem permite à aplicação de recepção que descubra o originador da mensagem. Ela contém, por exemplo, o nome do aplicativo que colocou a mensagem e o ID de usuário associados ao aplicativo.
- Quando um canal de mensagens se inicia, é possível que o agente de canal de mensagens (MCA) autentique seu parceiro em cada extremidade do canal. Essa técnica é conhecida como *autenticação mútua*. Para o MCA emitente, essa é a garantia de que o parceiro ao qual ele está prestes a enviar mensagens é genuíno. Para o MCA receptor, há uma garantia semelhante de que ele está prestes a receber mensagens de um parceiro genuíno.

Conceitos relacionados

“Identificação e autenticação” na página 5
Identificação é a capacidade de identificar exclusivamente um usuário de um sistema ou um aplicativo que esteja sendo executado no sistema. *Autenticação* é a capacidade de provar que um usuário ou um aplicativo é realmente quem essa pessoa ou o que esse aplicativo diz ser.

Autorização em IBM WebSphere MQ .

É possível usar a autorização para limitar o que indivíduos ou aplicativos específicos podem fazer em seu ambiente do IBM WebSphere MQ .

Aqui estão alguns exemplos de autorização em um ambiente do IBM WebSphere MQ :

- Permitindo que apenas um administrador autorizado emita comandos para gerenciar os recursos do IBM WebSphere MQ
- Permitir que um aplicativo se conecte a um gerenciador de filas somente se o ID de usuário associado ao aplicativo estiver autorizado a fazê-lo.
- Permitir que um aplicativo abra somente as filas que são necessárias para sua função.
- Permitir que um aplicativo assine apenas os tópicos que forem necessários para sua função.
- Permitir que um aplicativo execute apenas operações em uma fila que sejam necessárias à sua função. Por exemplo, é possível que um aplicativo necessite somente procurar mensagens em uma fila especifica, e não colocar ou receber mensagens.

Para obter mais informações sobre como configurar a autorização, consulte “Planejando a autorização” na página 50 e os subtópicos associados.

Conceitos relacionados

“Authorization” na página 6

A *autorização* protege os recursos críticos em um sistema, limitando o acesso somente a usuários autorizados e seus aplicativos. Ele previne o uso não autorizado de um recurso ou o uso de um recurso de maneira não autorizada.

Auditoria em IBM WebSphere MQ

O IBM WebSphere MQ pode emitir mensagens de eventos para registrar que uma atividade incomum ocorreu.

Aqui estão alguns exemplos de auditoria em um ambiente do IBM WebSphere MQ:

- Um aplicativo tenta abrir uma fila para que ele não está autorizado a abrir. Uma mensagem do evento de instrumentação é emitida. Inspecionando a mensagem do evento, você descobre que essa tentativa ocorreu e pode decidir qual ação é necessária.
- Um aplicativo tenta abrir um canal, mas a tentativa falhou porque o SSL não permite a conexão. Uma mensagem do evento de instrumentação é emitida. Inspecionando a mensagem do evento, você descobre que essa tentativa ocorreu e pode decidir qual ação é necessária.

Conceitos relacionados

"Auditing" na página 6
Auditoria é o processo de gravação e verificação de eventos para detectar se qualquer atividade inesperada ou desautorizada ocorreu, ou se nenhuma tentativa foi feita para executar essa atividade.

Confidencialidade em IBM WebSphere MQ

É possível implementar a confidencialidade no IBM WebSphere MQ, criptografando as mensagens.

Aqui estão alguns exemplos de como a confidencialidade pode ser assegurada em um ambiente do IBM WebSphere MQ :

- Depois que um MCA de envio recebe uma mensagem de uma fila de transmissão, o IBM WebSphere MQ usa SSL ou TLS para criptografar a mensagem antes de ela ser enviada pela rede ao MCA de recebimento. Na outra extremidade do canal, a mensagem é descodificada antes que o MCA receptor coloque-a em sua fila de destino.
- Enquanto as mensagens são armazenadas em uma fila local, os mecanismos de controle de acesso fornecidos pelo IBM WebSphere MQ podem ser considerados suficientes para proteger os seus conteúdos contra divulgação não autorizada. No entanto, para um maior nível de segurança, é possível usar o IBM WebSphere MQ Advanced Message Security para criptografar as mensagens armazenadas nas filas.

Conceitos relacionados

"Sigilosidade" na página 7
O serviço de *confidencialidade* protege informações sensitivas de exposições não autorizadas.

Integridade de dados no IBM WebSphere MQ ..

É possível usar um serviço de integridade de dados para detectar se uma mensagem foi modificada.

Aqui estão alguns exemplos de como a integridade de dados pode ser assegurada em um ambiente do IBM WebSphere MQ :

- É possível usar SSL ou TLS para detectar se o conteúdo de uma mensagem foi deliberadamente modificado enquanto era transmitido por uma rede. No SSL e TLS, o algoritmo de trecho da mensagem fornece detecção de mensagens modificadas em trânsito. Todos os IBM WebSphere MQ CipherSpecs fornecem um algoritmo de compilação de mensagens, exceto para TLS_RSA_WITH_NULL_NULL, que não fornece a integridade dos dados da mensagem.
- Enquanto as mensagens são armazenadas em uma fila local, os mecanismos de controle de acesso fornecidos pelo IBM WebSphere MQ podem ser considerados suficientes para evitar a modificação intencional do conteúdo das mensagens. No entanto, para obter um nível de segurança maior, é possível usar o IBM WebSphere MQ Advanced Message Security para detectar se o conteúdo de uma

mensagem foi intencionalmente modificado entre o momento que a mensagem foi colocada na fila e o momento em que foi recuperada.

Conceitos relacionados

“Integridade de dados” na página 7
O serviços *integridade dos dados* detecta se houve modificação não-autorizada dos dados.

Criptografia em IBM WebSphere MQ

O IBM WebSphere MQ fornece criptografia usando os protocolos Secure Sockets Layer (SSL) e Transport Security Layer (TSL).

Para obter informações adicionais, consulte “Protocolos de segurança no IBM WebSphere MQ” na página 23.

Conceitos relacionados

“Conceitos criptográficos” na página 7
Esta coleção de tópicos descreve os conceitos de criptografia aplicáveis ao WebSphere MQ

Protocolos de segurança no IBM WebSphere MQ

O IBM WebSphere MQ suporta os protocolos Transport Layer Security (TLS) e Secure Sockets Layer (SSL) para fornecer segurança em nível de link para canais de mensagens e canais de MQI.

Canais de mensagens e canais de MQI podem usar o protocolo SSL ou TLS para fornecer segurança no nível do link.Um MCA do responsável pela chamada é um cliente SSL ou TLS, e um MCA do respondente é um servidor SSL ou TLS. O WebSphere MQ suporta a Versão 3.0 do protocolo SSL e a Versão 1.0 e a Versão 1.2 do protocolo Segurança da Camada de Transporte (TLS). Especifique os algoritmos criptográficos que são usados pelo SSL ou protocolo fornecendo um CipherSpec como parte da definição de canal.

Em cada extremidade de um canal de mensagens, e na extremidade do servidor de um canal do MQI, o MCA atua em nome do gerenciador de filas ao qual está conectado. Durante o handshake do SSL ou TLS, o MCA envia o certificado digital do gerenciador de filas para o seu parceiro MCA na outra extremidade do canal. O código do WebSphere MQ na extremidade do cliente de um canal MQI age em nome do usuário do aplicativo cliente WebSphere MQ . Durante o handshake SSL ou TLS, o código WebSphere MQ envia o certificado digital do usuário para o MCA na extremidade do servidor do canal MQI.

Os gerenciadores de filas e os usuários do cliente WebSphere MQ não precisam ter certificados digitais pessoais associados a eles quando estiverem agindo como clientes SSL ou TLS, a menos que SSLCAUTH (REQUIRED) seja especificado no lado do servidor do canal.

Os certificados digitais são armazenados em um *repositório de chaves* O atributo do gerenciador de filas *SSLKeyRepository* especifica o local do repositório de chaves que contém o certificado digital do gerenciadorda fila. On a WebSphere MQ client system, the MQSSLKEYR environment variable specifies the location of the key repository that holds the user's digital certificate. Como alternativa, um aplicativo cliente WebSphere MQ pode especificar seu local no campo *KeyRepository* da estrutura de opções de configuração SSL e TLS, MQSCO, em uma chamada MQCONNX. Consulte os tópicos relacionados para obter mais informações sobre repositórios de chaves e como especificar onde eles estão localizados.

Conceitos relacionados

“Protocolos de Segurança Criptográficos: SSL e TLS” na página 14
Os protocolos de criptografia fornecem conexões seguras, permitindo que dois grupos de comuniquem com privacidade e integridade de dados. O protocolo Transport Layer Security (TLS) surgiu desse do Secure Sockets Layer (SSL). IBM WebSphere MQ suporta SSL e TLS.

IBM WebSphere MQ Suporte para SSL e TLS

O IBM WebSphere MQ suporta o protocolo Secure Sockets Layer (SSL) e o protocolo Transport Layer Security (TLS).

Para obter mais informações sobre os protocolos SSL e TLS, consulte as informações relacionadas..

O IBM WebSphere MQ fornece o suporte a seguir para o SSL Versão 3.0 e TLS 1.0 e TLS 1.2:

Clientes Java e JMS
Esses clientes utilizam JVM para fornecer suporte ao SSL e ao TLS.

Sistemas UNIX, Linux, and Windowse HP Integrity NonStop Server
Para sistemas UNIX, Linux, and Windowse HP Integrity NonStop Server , o suporte SSL e TLS é instalado com IBM WebSphere MQ.

Para obter informações sobre quaisquer pré-requisitos para o IBM WebSphere MQ suporte SSL e TLS, consulte Requisitos do sistema para IBM WebSphere MQ.

O repositório de chaves SSL ou TLS
Uma conexão SSL ou TLS mutuamente autenticada requer um repositório de chaves (que pode ser conhecido por nomes diferentes em plataformas diferentes) em cada extremidade da conexão... O repositório de chaves inclui certificados digitais e chaves privadas.

Estas informações usam o termo geral *repositório de chaves* para descrever o armazenamento para certificados digitais e suas chaves privadas associadas. Os nomes de loja específicos usados nas plataformas e ambientes que suportam SSL e TLS são:

Java e JMS	keystore e armazenamento confiável
Windows UNIX Linux Windows UNIX Linux Sistemas Windows , UNIX and Linux	chave do arquivo do banco de dados

Para obter mais informações, consulte "Certificados Digitais" na página 9 e "Conceitos do Secure Sockets Layer (SSL) e Transport Layer Security (TLS)" na página 14.

Uma conexão SSL ou TSL mutualmente autenticada requer um repositório de chaves em cada término da conexão. O repositório de chaves pode conter:

- Vários certificados de CA de várias Autoridades de Certificação que permitem que o gerenciador de filas ou o cliente verifique certificados recebidos de seu parceiro na extremidade remota da conexão. Certificados individuais podem estar em uma cadeia de certificados.
- Um ou mais certificados pessoais recebidos de uma Autoridade de Certificação. Você associa um certificado pessoal separado a cada gerenciador de filas ou cliente MQI do WebSphere MQ Certificados pessoais são essenciais em um cliente SSL ou TLS se a autenticação mútua for necessária. Se a autenticação mútua não for necessária, os certificados pessoais não serão necessários no cliente. O repositório de chaves também pode conter a chave privada correspondente a cada certificado pessoal.
- As solicitações de certificados que estão aguardando para serem assinadas por uma autoridade de certificação confiável.

Para obter mais informações sobre como proteger seu repositório de chaves, consulte "Protegendo repositórios de chaves do IBM WebSphere MQ" na página 25.

A localização do repositório de chaves depende da plataforma que você está utilizando:

Windows UNIX Linux **Windows, UNIX and Linux sistemas**
Em Windows, UNIX and Linux sistemas o repositório de chaves é um arquivo de banco de dados de chaves O nome do arquivo de banco de dados da chave deve ter uma extensão de arquivos `.kdb`. Por exemplo, no UNIX and Linux, o arquivo do banco de dados de chave padrão para o gerenciador de filas `QM1` é `/var/mqm/qmgrs/QM1/ssl/key.kdb` Se o IBM WebSphere MQ for instalado no local padrão, o caminho equivalente no Windows será `C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\key.kdb`

Nos sistemas Windows , UNIX and Linux , cada arquivo do banco de dados de chave possui um arquivo stash de senha associado Este arquivo mantém senhas codificadas que permitem que o programa acesse o banco de dados de chaves. O arquivo stash de senha deve estar no mesmo

diretório e ter a mesma raiz de arquivo que o banco de dados de chave e deve terminar com o sufixo `.sth` , por exemplo `/var/mqm/qmgrs/QM1/ssl/key.sth` .

Nota: Nos sistemas Windows, UNIX and Linux , as placas de hardware criptográficas PKCS #11 podem conter os certificados e as chaves que, de outra forma, são mantidos em um arquivo de banco de dados de chaves Quando certificados e chaves são retidos em placas PKCS #11 , o WebSphere MQ ainda requer acesso a um arquivo de banco de dados de chaves e a um arquivo stash de senha.

Nos sistemas Windows e UNIX , o banco de dados de chaves também contém a chave privada para o certificado pessoal associado ao gerenciador de filas ou cliente MQI do WebSphere MQ .

Protegendo repositórios de chaves do IBM WebSphere MQ
O repositório de chaves para o IBM WebSphere MQ é um arquivo Certifique-se de que somente o usuário pretendido possa acessar o arquivo repositório de chaves. Isso impede um intruso ou outro usuário não autorizado de copiar o arquivo repositório de chaves para outro sistema, e então configurar um ID de usuário idêntico naquele sistema para personificar o usuário pretendido.

As permissões sobre os arquivos dependem da umask do usuário e da ferramenta usada. No Windows, as contas do IBM WebSphere MQ requerem permissão `BypassTraverseChecking`, que significa que as permissões das pastas no caminho de arquivo não têm efeito.

Verifique as permissões dos arquivos de repositório de chaves e certifique-se de que os arquivos e pasta recipiente não sejam globalmente legíveis, de preferência, nem legíveis por grupos.

Seja qual for o sistema usado, é uma boa prática tornar o keystore somente leitura, apenas com o administrador tendo permissão de ativar operações de gravação para executar manutenção.

Na prática, você deve proteger todos os keystores, seja qual for a localização e se forem protegidos por senha ou não; proteja os repositórios de chaves.

Atualizando o repositório de chaves do gerenciador de filas
Ao mudar o conteúdo de um repositório de chaves, o gerenciador de filas não coleta imediatamente o novo conteúdo. Para um gerenciador de filas usar o novo conteúdo do repositório de chaves, deve-se emitir o comando REFRESH SECURITY TYPE(SSL).

Esse processo é intencional e evita a situação em que vários canais em execução poderia usar versões diferentes de um repositório de chaves. Como um controle de segurança, apenas uma versão de um repositório de chaves pode ser carregada pelo gerenciador de filas a qualquer momento.

Para obter mais informações sobre o comando REFRESH SECURITY TYPE(SSL), consulte REFRESH SECURITY.

Também é possível atualizar um repositório de chaves usando comandos PCF ou o WebSphere MQ Explorer. Para obter mais informações, consulte o comando MQCMD_REFRESH_SECURITY e o tópico *Atualizando a Segurança SSL ou TLS* na seção WebSphere MQ Explorer desta documentação do produto.

Conceitos relacionados
“Atualizando a Visualização de um Cliente do Conteúdo do Repositório de Chaves SSL e Configurações SSL” na página 25
Para atualizar o aplicativo cliente com o conteúdo atualizado do repositório de chaves, deve-se parar e reiniciar o aplicativo cliente.

Atualizando a Visualização de um Cliente do Conteúdo do Repositório de Chaves SSL e Configurações SSL
Para atualizar o aplicativo cliente com o conteúdo atualizado do repositório de chaves, deve-se parar e reiniciar o aplicativo cliente.

Não é possível atualizar a segurança em um cliente WebSphere MQ ; não há equivalente ao comando REFRESH SECURITY TYPE (SSL) para clientes (consulte REFRESH SECURITY) para mais informações.

Deve-se parar e reiniciar o aplicativo sempre que o certificado de segurança for mudado, para atualizar o aplicativo cliente com o conteúdo atualizado do repositório de chaves.

Se reiniciar o canal atualiza as configurações, e se o seu aplicativo tem reconexão lógica, é possível atualizar a segurança no cliente emitindo o comando STOP CHL STATUS(INACTIVE).

Conceitos relacionados

“Atualizando o repositório de chaves do gerenciador de filas” na página 25
Ao mudar o conteúdo de um repositório de chaves, o gerenciador de filas não coleta imediatamente o novo conteúdo. Para um gerenciador de filas usar o novo conteúdo do repositório de chaves, deve-se emitir o comando REFRESH SECURITY TYPE(SSL).

FIPS (Federal Information Processing Standards)

Este tópico apresenta o Federal Information Processing Standards (FIPS) Cryptomodule Validation Program do US National Institute of Standards and Technology e as funções criptográficas que podem ser usadas em canais SSL ou TLS, para sistemas Windows, UNIX and Linuxe z/OS .

A conformidade do FIPS 140-2 de uma conexão IBM WebSphere MQ SSL ou TLS em sistemas UNIX, Linuxe Windows é localizada aqui “Federal Information Processing Standards (FIPS) para UNIX, Linuxe Windows” na página 26

Se o hardware de criptografia estiver presente, os módulos de criptografia usados pelo IBM WebSphere MQ podem ser configurados para ser aqueles fornecidos pelo fabricante do hardware. Se isso for feito, a configuração só ficará em conformidade com o FIPS se esses módulos criptográficos forem certificados pelo FIPS.

Com o passar do tempo, os Federal Information Processing Standards são atualizados para refletirem novos ataques contra algoritmos de criptografia e protocolos. Por exemplo, alguns CipherSpecs podem deixar de ser certificados por FIPS. Quando tais mudanças ocorrem, o IBM WebSphere MQ também é atualizado para implementar o padrão mais recente. Como um resultado, você poderá ver as mudanças no comportamento após a aplicação da manutenção.

Conceitos relacionados

“Especificando que Apenas CipherSpecs Certificados por FIPS São Usados no Tempo de Execução no Cliente de MQI” na página 111
Crie seus repositórios de chaves usando software compatível com FIPS e, em seguida, especifique que o canal deve usar CipherSpecs certificados por FIPS.

“Usando iKeyman, iKeycmd, runmqakm e runmqckm” na página 116
Nos sistemas UNIX, Linux e Windows , gerencie chaves e certificados digitais com a GUI do iKeyman ou a partir da linha de comandos usando iKeycmd ou runmqakm.

Tarefas relacionadas

Ativando SSL em Classes do WebSphere MQ para Java

Usando Secure Sockets Layer (SSL) com classes do WebSphere MQ para JMS

Referências relacionadas

Propriedades SSL de Objetos JMS

Informações relacionadas

“Federal Information Processing Standards” na página 20
O governo dos EUA produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. O National Institute for Standards and Technology (NIST) é um órgão importante ligado a sistemas e segurança de TI. O NIST produz recomendações e normas, inclusive Federal Information Processing Standards (FIPS).

Federal Information Processing Standards (FIPS) para UNIX, Linuxe Windows

Quando a criptografia é necessária em um canal SSL ou TLS no Windows, UNIX and Linux sistemas, WebSphere MQ usa um pacote de criptografia chamado IBM Crypto for C (ICC). Nas plataformas Windows, UNIX and Linux , o software ICC passou pelo Federal Information Processing Standards (FIPS) Cryptomodule Validation Program do US National Institute of Standards and Technology, no nível 140-2.

A conformidade FIPS 140-2 de uma conexão SSL ou TLS do WebSphere MQ em sistemas Windows, UNIX and Linux é a seguinte:

- Para todos os canais de mensagens do IBM WebSphere MQ (exceto os tipos de canais CLNTCONN), a conexão será compatível com o FIPS se as seguintes condições forem atendidas:

  - A versão do ICC do GSKit instalada foi certificada como compatível com o FIPS 140-2 na versão do sistema operacional e arquitetura de hardware instaladas.
  - O atributo SSLFIPS do gerenciador de filas foi configurado para YES.
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.
- Para todos os aplicativos clientes MQI do IBM WebSphere MQ , a conexão usa GSKit e é compatível com FIPS se as condições a seguir forem atendidas:
  - A versão do ICC do GSKit instalada foi certificada como compatível com o FIPS 140-2 na versão do sistema operacional e arquitetura de hardware instaladas.
  - Você especificou que somente a criptografia certificada para FIPS deve ser usada, conforme descrito no tópico relacionado para o cliente de MQI.
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.
- Para classes IBM WebSphere MQ para aplicativos Java usando o modo cliente, a conexão usa as implementações SSL e TLS do JRE e é compatível com FIPS se as condições a seguir forem atendidas:
  - O Java Runtime Environment usado para executar o aplicativo é compatível com FIPS na versão do sistema operacional instalado e na arquitetura de hardware
  - Você especificou que apenas a criptografia certificada por FIPS deve ser usada, conforme descrito no tópico relacionado para o cliente Java
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.
- Para classes IBM WebSphere MQ para aplicativos JMS usando o modo cliente, a conexão usa as implementações SSL e TLS do JRE e é compatível com FIPS se as condições a seguir forem atendidas:
  - O Java Runtime Environment usado para executar o aplicativo é compatível com FIPS na versão do sistema operacional instalado e na arquitetura de hardware
  - Você especificou que apenas a criptografia certificada pelo FIPS deve ser usada, conforme descrito no tópico relacionado para o cliente do JMS
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.
- Para aplicativos clientes .NET não gerenciados, a conexão usa GSKit e é compatível com FIPS se as condições a seguir forem atendidas:
  - A versão do ICC do GSKit instalada foi certificada como compatível com o FIPS 140-2 na versão do sistema operacional e arquitetura de hardware instaladas.
  - Você especificou que apenas a criptografia certificada pelo FIPS deve ser usada, conforme descrito no tópico relacionado para o cliente .NET
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.
- Para aplicativos clientes XMS .NET não gerenciados, a conexão usa GSKit e é compatível com FIPS se as condições a seguir forem atendidas:
  - A versão do ICC do GSKit instalada foi certificada como compatível com o FIPS 140-2 na versão do sistema operacional e arquitetura de hardware instaladas.
  - Você especificou que apenas a criptografia certificada pelo FIPS deve ser usada, conforme descrito na documentação do XMS .NET
  - Todos os repositórios de chaves foram criados e manipulados usando somente software compatível com FIPS, como **runmqakm** com a opção `-fips`.

Todas as plataformas AIX, Linux, HP-UX, Solaris, Windowse z/OS suportadas são certificadas pelo FIPS 140-2, exceto conforme indicado no arquivo leia-me incluído com cada fix pack ou pacote de atualização

Para conexões SSL e TLS que usam o GSKit, o componente que é certificado pelo FIPS 140-2 é denominado *ICC*. É a versão desse componente que determina a conformidade com o FIPS do GSKit em qualquer plataforma fornecida. Para determinar a versão do ICC instalada atualmente, execute o comando **dspmqver -p 64 -v** ..

Aqui está um exemplo de extrato da saída **dspmqver -p 64 -v** relacionada ao ICC:

```
  ICC
 ============
@ (#) CompanyName: IBM Corporation
@(#)LegalTrademarks:  IBM
@ (#) FileDescription: IBM Crypto for C-language
@(#)FileVersion:      8.0.0.0
@(#)LegalCopyright:  Licensed Materials - Property of IBM
@(#)          ICC
@ (#) (C) Copyright IBM Corp. 2002, 2024.
@ (#) Todos os direitos reservados. US Government Users
@(#)          Restricted Rights - Use, duplication or disclosure
@(#)          restricted by GSA ADP Schedule Contract with IBM Corp.
@(#)ProductName:   icc_8.0 (GoldCoast Build) 100415
@(#)ProductVersion:   8.0.0.0
@(#)ProductInfo:      10/04/15.03:32:19.10/04/15.18:41:51
@(#)CMVCInfo:
```

A instrução de certificação NIST para o GSKit ICC 8 (incluído no GSKit 8) pode ser localizada no endereço a seguir: Cryptographic Module Validation Program.

Se o hardware de criptografia estiver presente, os módulos de criptografia usados pelo IBM WebSphere MQ podem ser configurados para ser aqueles fornecidos pelo fabricante do hardware. Se isso for feito, a configuração só ficará em conformidade com o FIPS se esses módulos criptográficos forem certificados pelo FIPS.

Nota: Clientes SSL e TLS do Solaris x86 de 32 bits configurados para operação compatível com FIPS 140-2 falham ao executar em sistemas Intel. Esta falha ocorre porque o arquivo de biblioteca GSKit-Crypto Solaris x86 32 bits compatível com FIPS 140-2 não carrega no Intel Chipset. Nos sistemas afetados, o erro AMQ9655 é relatado no log de erros do cliente. Para resolver esse problema, desative a conformidade com FIPS 140-2 ou recompile o aplicativo cliente de 64 bits porque o código de 64 bits não é afetado.

Restrições do Padrão de Criptografia de Dados triplo impostas ao operar em conformidade com o FIPS 140-2

Quando o WebSphere MQ é configurado para operar em conformidade com FIPS 140-2, restrições adicionais são impostas em relação ao Triple DES (3DES) CipherSpecs. Essas restrições permitem a conformidade com a recomendação do US NIST SP800-67.

1. Todas as partes do Padrão de Criptografia de Dados triplo devem ser exclusivas.
2. Nenhuma parte do Padrão de Criptografia de Dados triplo pode ser uma chave Weak, Semi-Weak ou Possibly-Weak, de acordo com as definições no NIST SP800-67.
3. Não mais que 32 GB de dados podem ser transmitidos através da conexão antes que uma reconfiguração de chave secreta deva ocorrer. Por padrão, o WebSphere MQ não reconfigura a chave de sessão secreta, portanto, essa reconfiguração deve ser configurada Falha ao ativar a reconfiguração da chave secreta ao usar um CipherSpec do Padrão de Criptografia de Dados triplo e resultados de conformidade com o FIPS 140-2 no fechamento da conexão com o erro AMQ9288 após a contagem máxima de bytes exceder. Para obter informações sobre como definir a reconfiguração de chave secreta, consulte “Reconfigurando as chaves secretas SSL e TLS” na página 227.

WebSphere MQ gera chaves de sessão Triple DES que já estão em conformidade com as regras 1 e 2. No entanto, para atender a terceira restrição, deve-se ativar a reconfiguração de chave secreta ao usar as CipherSpecs do Triple DES em uma configuração 140-2 do FIPS. Como alternativa, é possível evitar o uso do Padrão de Criptografia de Dados triplo.

Conceitos relacionados

“Especificando que Apenas CipherSpecs Certificados por FIPS São Usados no Tempo de Execução no Cliente de MQI” na página 111

Crie seus repositórios de chaves usando software compatível com FIPS e, em seguida, especifique que o canal deve usar CipherSpecs certificados por FIPS.

"Usando iKeyman, iKeycmd, runmqakm e runmqckm" na página 116
Nos sistemas UNIX, Linux e Windows , gerencie chaves e certificados digitais com a GUI do iKeyman ou a partir da linha de comandos usando iKeycmd ou runmqakm.

Tarefas relacionadas

Ativando SSL em Classes do WebSphere MQ para Java

Usando Secure Sockets Layer (SSL) com classes do WebSphere MQ para JMS

Referências relacionadas

Propriedades SSL de Objetos JMS

Informações relacionadas

"Federal Information Processing Standards" na página 20
O governo dos EUA produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. O National Institute for Standards and Technology (NIST) é um órgão importante ligado a sistemas e segurança de TI. O NIST produz recomendações e normas, inclusive Federal Information Processing Standards (FIPS).

SSL e TLS no cliente MQI do IBM WebSphere MQ

O IBM WebSphere MQ suporta SSL e TLS em clientes. É possível customizar o uso de SSL ou de TLS de várias maneiras.

O IBM WebSphere MQ fornece suporte SSL e TLS para clientes do IBM WebSphere MQ MQI em sistemas Windows, UNIX and Linux Se você estiver usando classes IBM WebSphere MQ para Java, consulte Usando classes WebSphere MQ para Java e se estiver usando classes IBM WebSphere MQ para JMS, consulte Usando classes WebSphere MQ para JMS. O restante desta seção não se aplica aos ambientes Java ou JMS..

É possível especificar o repositório de chave para um cliente MQI do IBM WebSphere MQ com o valor MQSSLKEYR em seu arquivo de configuração do cliente IBM WebSphere MQ ou quando seu aplicativo faz uma chamada MQCONNX Você tem três opções para especificar que um canal utilize o SSL:

- Utilizando uma tabela de definição de canais
- Usando a estrutura de opções de configuração do SSL, MQSCO, ou uma chamada de MQCONNX
- Usando o Active Directory (em sistemas Windows)

Você não pode utilizar a variável de ambiente MQSERVER para especificar que um canal utilize o SSL.

É possível continuar executando seus aplicativos clientes MQI existentes do IBM WebSphere MQ sem SSL, desde que o SSL não seja especificado na outra extremidade do canal.

Se forem feitas mudanças em um computador cliente no conteúdo do Repositório de Chaves SSL, na localização do Repositório de Chaves SSL, nas Informações de Autenticação, ou nos parâmetros do Hardware de criptografia, será necessário terminar todas as conexões SSL para que essas mudanças sejam refletidas nos canais de conexão do cliente que o aplicativo está usando para se conectar ao gerenciador de filas. Depois que todas as conexões tiverem terminado, reinicie os canais SSL. Todos as novas configurações de SSL serão utilizadas. Essas configurações são análogas àquelas atualizadas pelo comando REFRESH SECURITY TYPE(SSL) em sistemas do Gerenciadores de Filas.

When your IBM WebSphere MQ MQI client runs on a Janelas, UNIX and Linux system with cryptographic hardware, you configure that hardware with the MQSSLCRYP environment variable. Esta variável é equivalente ao parâmetro SSLCRYP no comando ALTER QMGR MQSC. Consulte ALTER QMGR para obter uma descrição do parâmetro SSLCRYP no comando ALTER QMGR MQSC. Se você usar a versão GSK_PCS11 do parâmetro SSLCRYP, o rótulo do token PKCS #11 deverá ser especificado inteiramente em minúsculas.

A reconfiguração da chave secreta SSL e o FIPS são suportados nos clientes MQI do IBM WebSphere MQ Para obter mais informações, consulte o "Reconfigurando as chaves secretas SSL e TLS" na página 227 e o "Federal Information Processing Standards (FIPS) para UNIX, Linuxe Windows" na página 26.

Consulte “Configurando a segurança do cliente MQI do IBM WebSphere MQ” na página 110 para obter mais informações sobre o suporte SSL para clientes MQI IBM WebSphere MQ .

Tarefas relacionadas

Configurando um Cliente Usando um Arquivo de Configuração

Especificando que um Canal MQI Use SSL

Para um canal de MQI usar SSL, o valor do atributo *SSLCipherSpec* do canal de conexão do cliente deve ser o nome de um CipherSpec que é suportado pelo IBM WebSphere MQ na plataforma do cliente.

É possível definir um canal de conexão do cliente com um valor para este atributo das seguintes maneiras. Eles são listados na ordem de precedência decrescente.

1. Quando uma saída PreConnect fornece uma estrutura de definição de canal para uso.

   Uma saída PreConnect pode fornecer o nome de um CipherSpec no campo *SSLCipherSpec* de uma estrutura de definição de canal, MQCD. Esta estrutura é retornada no campo **ppMQCDArrayPtr** da estrutura do parâmetro de saída MQNXP usada pela saída PreConnect.
2. Quando um aplicativo cliente MQI do WebSphere MQ emite uma chamada MQCONNX.

   O aplicativo pode especificar o nome de um CipherSpec no campo *SSLCipherSpec* de uma estrutura de definição de canal, MQCD.Esta estrutura é referenciada pela estrutura de opções de conexão, MQCNO, que é um parâmetro na chamada MQCONNX.
3. Usando uma Tabela de Definição de Canal de Cliente (CCDT).

   Uma ou mais entradas em uma tabela de definição de canal do cliente podem especificar o nome de um CipherSpec.Por exemplo, se você criar uma entrada usando o comando DEFINE CHANNEL MQSC, poderá usar o parâmetro SSLCIPH no comando para especificar o nome de um CipherSpec.
4. Usando o Active Directory no Windows

   Em sistemas Windows , é possível usar o comando de controle **setmqscp** para publicar as definições de canal de conexão do cliente no Active Directory.. Uma ou mais destas definições podem especificar o nome de um CipherSpec.

Por exemplo, se um aplicativo cliente fornecer uma definição de canal de conexão do cliente em uma estrutura MQCD em uma chamada MQCONNX, essa definição será usada em preferência a quaisquer entradas em uma tabela de definições de canal do cliente que possam ser acessadas pelo cliente do WebSphere MQ

Não é possível usar a variável de ambiente MQSERVER para fornecer a definição de canal na extremidade do cliente de um canal MQI que usa SSL.

Para verificar se um certificado de cliente fluiu, exiba o status do canal na extremidade do servidor de um canal para a presença de um valor de parâmetro de nome de mesmo nível.

Conceitos relacionados

“Especificando um CipherSpec para um cliente MQI do IBM WebSphere MQ” na página 227
Você tem três opções para especificar um CipherSpec para um cliente MQI do IBM WebSphere MQ

CipherSpecs e CipherSuites em IBM WebSphere MQ

O IBM WebSphere MQ suporta SSL e TLS CipherSpecse algoritmos RSA e Diffie-Hellman.

O WebSphere MQ suporta SSL V3 e TLS V1.0 e V1.2 CipherSpecs.

O WebSphere MQ suporta os algoritmos de troca de chave e autenticação RSA e Diffie-Hellman. O tamanho da chave usada durante o handshake SSL pode depender do certificado digital usado, mas alguns CipherSpecs incluem uma especificação do tamanho da chave de handshake. Tamanhos maiores de chaves de handshake fornecem autenticação mais consistente. Com tamanhos de chaves menores, o protocolo de reconhecimento é mais veloz.

Conceitos relacionados

“CipherSpecs e CipherSuites” na página 18

Os protocolos de segurança criptográficos devem concordar sobre os algoritmos usados por uma conexão segura. CipherSpecs e CipherSuites definem combinações específicas de algoritmos.

Criptografia do Conjunto B da NSA no IBM WebSphere MQ
Este tópico fornece informações sobre como configurar IBM WebSphere MQ em sistemas Windows, Linuxe UNIX para conformidade com o perfil do TLS 1.2 compatível com o Suite B.

Com o tempo, o Padrão do Conjunto B de Criptografia da NSA é atualizado para refletir novos ataques contra algoritmos de criptografia e protocolos. Por exemplo: alguns CipherSpecs podem deixar de serem certificados pelo Conjunto B. Quando tais mudanças ocorrem, o IBM WebSphere MQ também é atualizado para implementar o padrão mais recente. Como um resultado, você poderá ver as mudanças no comportamento após a aplicação da manutenção. O arquivo leia-me do IBM WebSphere MQ Version 7.5 lista a versão do Conjunto B cumprido por nível de manutenção do produto. Se você configurar o IBM WebSphere MQ para cumprir a conformidade do Conjunto B, sempre consulte o arquivo leia-me ao planejar aplicar a manutenção (consulte IBM MQ, WebSphere MQe READMEs do produto MQSeries).

Nos sistemas Windows, UNIXe Linux , IBM WebSphere MQ pode ser configurado para estar em conformidade com o perfil TLS 1.2 compatível com o Suite B nos níveis de segurança mostrados na Tabela 1.

Tabela 1. Níveis de Segurança do Conjunto B com CipherSpecs Permitidos e Algoritmos de Assinatura Digital

Níveis de segurança	**CipherSpecs Permitidos**	**Algoritmos de assinatura digital**
128 bits	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA com SHA-256 ECDSA com SHA-384
192 bits	ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA com SHA-384
Ambos [1]	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA com SHA-256 ECDSA com SHA-384

1. É possível configurar ambos os níveis de segurança de 128 e 192 bits simultaneamente. Como a configuração do Conjunto B determina os algoritmos criptográficos mínimos aceitáveis, a configuração de ambos os níveis de segurança é equivalente a configurar apenas o nível de segurança de 128 bits. Os algoritmos criptográficos do nível de segurança de 192 bits são mais fortes do que o mínimo requerido para o nível de segurança de 128 bits, portanto, eles são permitidos para o nível de segurança de 128 bits, mesmo se o nível de segurança 192 bits não estiver ativado.

Nota: As convenções de nomenclatura usadas para o nível de segurança não representam necessariamente o tamanho da curva elíptica ou o tamanho da chave do algoritmo de criptografia AES.

configuração do Conjunto B para CipherSpec

Embora o comportamento padrão do IBM WebSphere MQ não esteja em conformidade com o padrão do Conjunto B, o IBM WebSphere MQ pode ser configurado para estar em conformidade com um ou ambos os níveis de segurança nos sistemas Windows, UNIX e Linux .. Após a configuração bem-sucedida do IBM WebSphere MQ para usar com o Conjunto B, qualquer tentativa de iniciar um canal de saída usando um CipherSpec que não seja compatível com o Conjunto B resulta no erro AMQ9282. Esta atividade também resulta no retorno do código de razão MQRC_CIPHER_SPEC_NOT_SUITE_B por parte do cliente do MQI. Assim como tentar iniciar um canal de entrada usando um CipherSpec que não esteja em conformidade com os resultados de configuração do Conjunto B no erro AMQ9616.

Para obter mais informações sobre o WebSphere MQ CipherSpecs, consulte “Especificando CipherSpecs” na página 221

Conjunto B e Certificados Digitais

O Conjunto B restringe os algoritmos de assinatura digital que podem ser usados para assinar certificados digitais. Conjunto B também restringe o tipo de chave pública que certificados pode conter. Portanto, WebSphere MQ deve ser configurado para usar certificados cujo algoritmo de assinatura digital e tipo de chave pública são permitidos pelo nível de segurança configurado do Suite B do parceiro remoto. Certificados digitais que não cumprirem os requisitos de nível de segurança serão rejeitados e a conexão falhará com o erro AMQ9633 ou AMQ9285.

Para o nível de segurança do Conjunto B de 128 bits, a chave pública do assunto do certificado é necessária para usar a curva elíptica P-256 NIST ou a curva elíptica P-384 NIST e ser assinada com a curva elíptica P-256 NIST ou a curva elíptica P-384 NIST. No nível de segurança de 192 bits do Conjunto B, a chave pública do assunto do certificado deve usar e ser assinada pela curva elíptica NIST P-384.

Para obter um certificado adequado a operações compatíveis com o Conjunto B, use o comando **runmqakm** e especifique o parâmetro **-sig_alg** para solicitar um algoritmo de assinatura digital apropriado. Os valores de parâmetro EC_ecdsa_with_SHA256 e EC_ecdsa_with_SHA384 **-sig_alg** correspondem às chaves de curva elíptica assinadas pelos algoritmos de assinatura digital permitidos peloConjunto B.

Para obter mais informações sobre o comando **runmqakm**, consulte opções runmqckm e runmqakm.

Nota: As ferramentas **iKeycmd** e **iKeyman** não suportam a criação de certificados digitais para operações compatíveis com o Conjunto B.

Criando e Solicitando Certificados Digitais

Para criar um certificado digital autoassinado para testes com o Conjunto B, consulte “Criando um certificado pessoal auto-assinado em sistemas UNIX, Linux, and Windows” na página 124

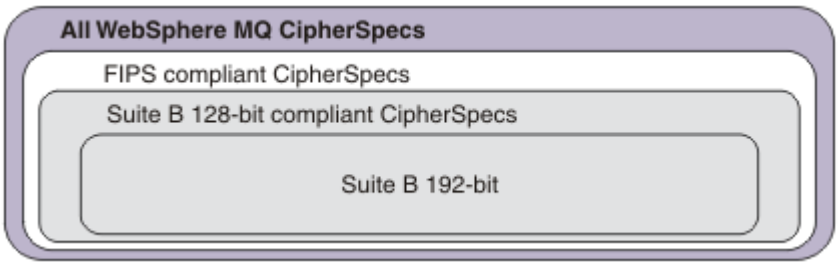
Para solicitar um certificado digital assinado pela CA para uso com o Conjunto B, consulte “Solicitando um certificado pessoal nos sistemas UNIX, Linux, and Windows” na página 127.

Nota: A autoridade de certificação que está sendo usada deve gerar os certificados digitais que satisfazem os requisitos descritos em IETF RFC 6460.

FIPS 140-2 e Conjunto B

O padrão do Conjunto B é conceitualmente semelhante a FIPS 140-2, já que restringe o conjunto de algoritmos criptográficos ativados para fornecer um nível de garantia de segurança. Os CipherSpecs do Conjunto B atualmente suportados podem ser usados quando o IBM WebSphere MQ é configurado para operação em conformidade com FIPS 140-2. Portanto, é possível configurar o WebSphere MQ para conformidade FIPS e Suite B simultaneamente, nesse caso, ambos os conjuntos de restrições se aplicam.

O diagrama a seguir ilustra o relacionamento entre esses subconjuntos:



Configurando a operação compatível com o WebSphere MQ para Suite B

Para obter informações sobre como configurar IBM WebSphere MQ no Windows, UNIX e Linux para operação compatível com o Suite B, consulte “Configurando o IBM WebSphere MQ para o Conjunto B” na página 33.

O IBM WebSphere MQ não suporta a operação compatível com o Conjunto B nas plataformas IBM i e z/OS Os clientes Java e JMS do WebSphere MQ também não suportam a operação compatível com o Suite B.

Conceitos relacionados

“Especificando que Apenas CipherSpecs Certificados por FIPS São Usados no Tempo de Execução no Cliente de MQI” na página 111
Crie seus repositórios de chaves usando software compatível com FIPS e, em seguida, especifique que o canal deve usar CipherSpecs certificados por FIPS.

Configurando o IBM WebSphere MQ para o Conjunto B
IBM WebSphere MQ pode ser configurado para operar em conformidade com o padrão NSA Suite B em sistemas UNIX, Linux, and Windows .

O Conjunto B restringe o conjunto de algoritmos criptográficos ativados, a fim de fornecer um nível de segurança seguro.IBM WebSphere MQ pode ser configurado para operar em conformidade com o Suite B para fornecer um nível aprimorado de segurança Para obter informações adicionais sobre o Conjunto B, consulte “Criptografia do Conjunto B da Agência Nacional de Segurança (NSA)” na página 20. Para obter mais informações sobre a configuração do Conjunto B e seu efeito sobre os canais SSL e TLS, consulte “Criptografia do Conjunto B da NSA no IBM WebSphere MQ” na página 31.

Gerenciador de Filas

Para um gerenciador de filas, use o comando **ALTER QMGR** com o parâmetro **SUITEB** para configurar os valores apropriados para o nível de segurança requerido. Para obter informações adicionais, consulte ALTER QMGR.

Também é possível usar o comando PCF **MQCMD_CHANGE_Q_MGR** com o parâmetro **MQIA_SUITE_B_STRENGTH** para configurar o gerenciador de fila para a operação compatível com Suite B

Cliente MQI

Por padrão, os clientes MQI não aplicam a conformidade do Conjunto B. É possível ativar o cliente MQI para conformidade do Conjunto B executando uma das opções abaixo:

1. Configurando o campo **EncryptionPolicySuiteB** na estrutura MQSCO em uma chamada MQCONNX para um ou mais dos valores abaixo:
   - MQ_SUITE_B_NONE
   - MQ_SUITE_B_128_BIT
   - MQ_SUITE_B_192_BIT

   Usar MQ_SUITE_B_NONE com qualquer outro valor é inválido.
2. Configurando a variável de ambiente MQSUITEB para um ou mais dos valores abaixo:
   - NENHUMA
   - 128_BIT
   - 192_BIT

   É possível especificar vários valores usando uma lista separada por vírgula. Usar o valor NONE com qualquer outro valor é inválido.
3. Configurando o atributo **EncryptionPolicySuiteB** na sub-rotina SSL do arquivo de configuração do cliente MQI para um ou mais dos valores abaixo:
   - NENHUMA
   - 128_BIT
   - 192_BIT

   É possível especificar vários valores usando uma lista separada por vírgula. Usar NONE com qualquer outro valor é inválido.

Nota: As configurações do cliente de MQI são listadas em ordem de prioridade. A estrutura de MSCO na chamada MQCONNX substitui a configuração na variável de ambiente MQSUITEB, que substitui o atributo na sub-rotina de SSL.

Para obter detalhes completos da estrutura de MQSCO, consulte MQSCO - Opções de configuração SSL.

Para obter mais informações sobre o uso do Conjunto B no arquivo de configuração do cliente, consulte a sub-rotina SSL do arquivo de configuração do cliente

Para obter informações adicionais sobre o uso da variável de ambiente MQSUITEB, consulte Variáveis de ambiente.

.REDE

Para clientes não gerenciados do .NET, a propriedade **MQC.ENCRYPTION_POLICY_SUITE_B** indica o tipo de segurança do Conjunto B necessária

Para obter informações sobre o uso do Conjunto B em classes IBM WebSphere MQ para .NET, consulte Classe MQEnvironment .NET

Políticas de validação de certificado no IBM WebSphere MQ
A política de validação de certificado determina com qual precisão a validação de cadeia de certificados está em conformidade com os padrões de segurança do segmento de mercado.

A política de validação de certificado depende da plataforma e do ambiente, como a seguir:

- Para aplicativos Java e JMS em todas as plataformas, a política de validação de certificado depende do componente JSSE do Java Runtime Environment. Para obter informações adicionais sobre a política de validação de certificado, consulte a documentação do seu JRE.
- Para sistemas UNIX, Linux, and Windows, a política de validação de certificado é fornecida pelo GSKit e pode ser configurada. Duas políticas de validação de certificado diferentes são suportadas:
  - Uma política de validação de certificado de legado, usada para máxima compatibilidade reversa e interoperabilidade com certificados digitais antigos que não estão em conformidade com os padrões de validação de certificado IETF atuais. Esta política é conhecida como a política Básica.
  - Uma política de validação de certificado rígida e em conformidade com padrões que impinge o padrão RFC 5280. Esta política é conhecida como a política Padrão.

Para obter informações sobre como configurar a política de validação de certificado em sistemas UNIX, Linux, and Windows, consulte “Configurando políticas de validação de certificado no IBM WebSphere MQ” na página 34. Para obter mais informações sobre as diferenças entre as políticas de validação de certificado Básico e Padrão, consulte Validação de certificado e design de política de confiança em UNIX, Linux e sistemas Windows

Configurando políticas de validação de certificado no IBM WebSphere MQ
Você pode especificar qual política de validação de certificado SSL/TLS é usada para validar certificados digitais recebidos de sistemas de parceiros remotos em quatro maneiras.

No gerenciador de filas, a política de validação de certificado pode ser definida das seguintes maneiras:

- Usando o atributo do gerenciador de filas *CERTVPOL*. Para obter mais informações sobre como configurar esse atributo, consulte ALTER QMGR

No cliente, existem vários métodos que podem ser usados para definir a política de validação de certificado. Se mais de um método é usado para definir a política, o cliente usará as configurações na seguinte ordem de prioridade:

1. Usando o campo *CertificateValPolicy* na estrutura MQSCO do cliente. Para obter informações adicionais sobre o uso desse campo consulte MQSCO - Opções de configuração SSL.
2. Usando a variável de ambiente do cliente, *MQCERTVPOL*. Para obter mais informações sobre como usar essa variável, consulte MQCERTVPOL .

3. Usando a configuração para o parâmetro de ajuste da sub-rotina SSL do cliente, *CertificateValPolicy*. Para obter mais informações sobre como usar essa configuração, consulte Sub-rotina SSL do arquivo de configuração do cliente

Para obter informações adicionais sobre as políticas de validação de certificado, consulte “Políticas de validação de certificado no IBM WebSphere MQ” na página 34.

Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ
Este tópico fornece informações sobre como escolher os certificados digitais e do CipherSpecs para sua política de segurança, delineando o relacionamento entre os certificados digitais e de CipherSpecs no IBM WebSphere MQ.

Em liberações anteriores do IBM WebSphere MQ, todos os CipherSpecs SSL e TLS suportados usavam o algoritmo RSA para assinaturas digitais e o contrato de chaves Todos os tipos suportados de certificado digital eram compatíveis com todos os CipherSpecs suportados, portanto, é possível mudar o CipherSpec para qualquer canal, sem precisar mudar os certificados digitais.

No IBM WebSphere MQ v7.5 , somente um subconjunto do CipherSpecs suportado pode ser usado com todos os tipos suportados de certificado digital Portanto, é necessário escolher um CipherSpec apropriado para o seu certificado digital. Da mesma forma, se a política de segurança de sua organização requer o uso de um CipherSpec específico, deve-se obter um certificado digital apropriado para esse CipherSpec.

Os algoritmos de assinatura digital MD5 e o TLS 1.2

Os certificados digitais assinados usando o algoritmo MD5 são rejeitados quando o protocolo TLS 1.2 é usado. Isso é porque o algoritmo MD5 é agora considerado fraco por muitos analistas de criptografia, e seu uso é normalmente desencorajado. Se desejar usar CipherSpecs mais recentes com base no protocolo TLS 1.2, assegure-se de que os certificados digitais não usem o algoritmo MD5 em suas assinaturas digitais. Os CipherSpecs antigos que usam os protocolos SSL 3.0 e TLS 1.0 não estão sujeitos a essa restrição e podem continuar a usar certificados com assinaturas digitais de MD5.

Para visualizar o algoritmo de assinatura digital para um certificado específico, é possível usar o comando **runmqakm** :

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

em que `cert_label` é o rótulo do certificado do algoritmo de assinatura digital que você precisa exibir

Nota: Embora a ferramenta **iKeycmd (runmqckm)** e a GUI do **iKeyman (strmqikm)** possam ser usadas para visualizar uma seleção de algoritmos de assinatura digital, a ferramenta **runmqakm** fornece um intervalo mais amplo.

A execução do comando **runmqakm** produzirá saída exibindo o uso do algoritmo de assinatura especificado:

```
Label : ibmwebspheremqexample
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
```

A linha `Signature Algorithm` mostra que o algoritmo `MD5WithRSASignature` é usado.. Este algoritmo é baseado em MD5 e, portanto, este certificado digital não pode ser usado com o CipherSpecs do TLS 1.2.

Interoperabilidade da curva elíptica e do RSA de CipherSpecs

Nem todos os CipherSpecs podem ser usados com todos os certificados digitais. Há três tipos de CipherSpec, indicados pelo prefixo de nome do CipherSpec. Cada tipo de CipherSpec impõe restrições diferentes do tipo de certificado digital que pode ser usado. Essas restrições se aplicam a todas as conexões SSL e TLS do WebSphere MQ , mas são particularmente relevantes para usuários da criptografia Elliptic Curve.

Os relacionamentos entre CipherSpecs e certificados digitais são resumidos na tabela a seguir:

Tabela 2. Relacionamentos entre os certificados digital e de CipherSpecs

Tipo	Prefixo de nome do CipherSpec	Descrição	Tipo de chave pública requerido	Algoritmo de criptografia de assinatura digital	Método de estabelecimento de chave secreta
1	ECDHE_ECDSA_	Os CipherSpecs que usam as chaves públicas da Curva Elíptica, as chaves secretas da Curva Elíptica e os algoritmos de assinatura digital da Curva Elíptica.	Curva Elíptica	ECDSA	ECDHE
2	ECDHE_RSA_	Os CipherSpecs que usam chaves públicas do RSA, chaves secretas da Curva Elíptica e algoritmos de assinatura digital da Curva Elíptica.	RSA	RSA	ECDHE

Tabela 2. Relacionamentos entre os certificados digital e de CipherSpecs (continuação)

Tipo	Prefixo de nome do CipherSpec	Descrição	Tipo de chave pública requerido	Algoritmo de criptografia de assinatura digital	Método de estabelecimento de chave secreta
3	(Todos os outros)	Os CipherSpecs que usam chaves públicas de RSA e algoritmos de assinatura digital de RSA.	RSA	RSA	RSA

Nota: Os CipherSpecs do tipo 1 e 2 são suportados apenas por gerenciadores de fila e clientes MQI do WebSphere MQ nas plataformas UNIX, Linux, and Windows .

A coluna de tipo de chave pública necessária mostra o tipo de chave pública que o certificado pessoal deve ter ao usar cada tipo de CipherSpec. O certificado pessoal é o certificado de entidade final que identifica o gerenciador de filas ou cliente para seu parceiro remoto.

O algoritmo de criptografia de assinatura digital se refere ao algoritmo de criptografia usado para validar o peer. O algoritmo de criptografia é usado juntamente com um algoritmo hash, como MD5, SHA-1 ou SHA-256 para calcular a assinatura digital. Há vários algoritmos de assinatura digital que podem ser usados, por exemplo, "RSA com MD5" ou "ECDSA com SHA-256". Na tabela, ECDSA refere-se ao conjunto de algoritmos de assinatura digital que usam ECDSA; RSA refere-se ao conjunto de algoritmos de assinatura digital que usam RSA. Qualquer algoritmo de assinatura digital suportado no conjunto pode ser usado, contanto que seja baseado no algoritmo de criptografia indicado.

Os CipherSpecs do tipo 1 requerem que o certificado pessoal tenha uma chave pública da Curva Elíptica. Quando estes CipherSpecs são usados, o acordo da chave efêmera de Diffie Hellman de Curva Elíptica é usado para estabelecer a chave secreta para a conexão.

Os CipherSpecs do tipo 2 requerem que o certificado pessoal tenha uma chave pública de RSA. Quando estes CipherSpecs são usados, o acordo da chave efêmera de Diffie Hellman de Curva Elíptica é usado para estabelecer a chave secreta para a conexão.

Os CipherSpecs do tipo 3 requerem que o certificado pessoal tenha uma chave pública de RSA. Quando estes CipherSpecs são usados, a troca de chave de RSA é usada para estabelecer a chave secreta para a conexão.

Esta lista de restrições não é completa: dependendo da configuração, pode haver restrições adicionais que podem afetar ainda mais a capacidade de interoperar. Por exemplo, se o WebSphere MQ for configurado para estar em conformidade com os padrões FIPS 140-2 ou NSA Suite B B, isso também limitará o intervalo de configurações permitidas Consulte a seção seguinte para obter informações adicionais.

Um gerenciador de filas do WebSphere MQ pode usar apenas um único certificado pessoal para se identificar. Isso significa que todos os canais no gerenciador de fila usarão o mesmo certificado digital e, portanto, cada gerenciador de fila poderá usar apenas um tipo de CipherSpec por vez Da mesma forma, um aplicativo cliente WebSphere MQ pode usar apenas um único certificado pessoal para se identificar. Isso significa que todas as conexões SSL e TLS dentro de um único processo de aplicativo usarão o mesmo certificado digital e, portanto, cada processo de aplicativo cliente poderá usar apenas um tipo de CipherSpec por vez

Os três tipos de CipherSpec não interoperam diretamente: esta é uma limitação dos padrões de SSL e TLS atuais. Por exemplo, suponha que você tenha escolhido usar o CipherSpec ECDHE_ECDSA_AES_128_CBC_SHA256 para um canal receptor denominado TO.QM1 em um gerenciador de filas denominado QM1. ECDHE_ECDSA_AES_128_CBC_SHA256 é um tipo 1 CipherSpec, portanto, QM1 deve ter um certificado pessoal com uma chave de Curva Elíptica e uma assinatura digital baseada em ECDSA. Todos os clientes e outros gerenciadores de filas que se comunicam diretamente com QM1

devem, portanto, ter certificados digitais que satisfaçam os requisitos CipherSpec do Tipo 1. Outros canais que se conectem ao gerenciador de filas QM1 podem usar outros CipherSpecs (por exemplo, ECDHE_ECDSA_3DES_EDE_CBC_SHA256), mas eles podem usar apenas o Tipo 1 CipherSpecs para se comunicar com QM1.

Ao planejar suas redes do WebSphere MQ , considere cuidadosamente quais canais requerem SSL ou TLS e assegure que todos os clientes e gerenciadores de filas que precisam interoperar usem o mesmo tipo de CipherSpecs e certificados digitais apropriados. Os padrões de IETF RFC 4492, RFC 5246 e RFC 6460 descrevem o uso detalhado de Elliptic Curve CipherSpecs no TLS 1.2

Para visualizar o algoritmo de assinatura digital e o tipo de chave pública de um certificado digital, é possível usar o comando **runmqakm**:

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

em que `cert_label` é o rótulo do certificado cujo algoritmo de assinatura digital precisa ser exibido.

A execução do comando **runmqakm** produzirá saída que exibe o Tipo de Chave Pública:

```
Label : ibmwebspheremqexample
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
```

A linha Tipo de Chave Pública neste caso mostra que o certificado tem uma chave pública de Curva Elíptica. A linha de Algoritmo de Assinatura neste caso mostra que o algoritmo EC_ecdsa_with_SHA384 está em uso: isso é baseado no algoritmo de ECDSA. Esse certificado é, portanto, adequado apenas para uso com o CipherSpecs tipo 1.

Também é possível usar a ferramenta do **iKeycmd (runmqckm)** com os mesmos parâmetros Além disso, a GUI **iKeyman (strmqikm)** pode ser usada para visualizar os algoritmos de assinatura digital, se você abrir o repositório de chaves e der um clique duplo no rótulo do certificado. No entanto, é aconselhado o uso da ferramenta **runmqakm** para visualizar certificados digitais, pois ela suporta uma ampla variedade de algoritmos.

Curva Elíptica de CipherSpecs e Conjunto B da NSA

Quando o WebSphere MQ é configurado para estar em conformidade com o perfil TLS 1.2 compatível com o Suite B, os algoritmos de CipherSpecs e de assinatura digital são restritos conforme descrito em

"Criptografia do Conjunto B da NSA no IBM WebSphere MQ" na página 31. Além disso, o intervalo de chaves aceitável do Elliptic Curve é reduzido de acordo com os níveis de segurança configurados.

No nível de segurança do Conjunto B de 128 bits, a chave pública do assunto do certificado é necessária para usar a curva elíptica de NIST P-256 ou NIST P-384 e ser assinada com a curva elíptica NIST P-256 ou NIST P-384. O comando **runmqakm** pode ser usado para solicitar certificados digitais para esse nível de segurança, usando um parâmetro -sig_alg de EC_ecdsa_with_SHA256 ou EC_ecdsa_with_SHA384.

No nível de segurança do Conjunto B de 192 bits, a chave pública do assunto do certificado é necessária para usar a curva elíptica de NIST P-384 e ser assinada com a curva elíptica NIST P-384. O comando **runmqakm** pode ser usado para solicitar certificados digitais para esse nível de segurança usando um parâmetro -sig_alg de EC_ecdsa_with_SHA384.

As curva elípticas de NIST suportadas são as seguintes:

Tabela 3. Curvas elípticas de NIST suportadas

Nome da curva NIST FIPS 186-3	**Nome da curva RFC 4492**	**Tamanho da chave da Curva Elíptica (bits)**
P-256	secp256r1	256
P-384	secp384r1	384
P-521	secp521r1	521

Nota: A curva elíptica NIST P-521 não pode ser usado para operação compatível com o Conjunto B.

Conceitos relacionados

"Especificando CipherSpecs" na página 221
Especifique um CipherSpec usando o parâmetro **SSLCIPH** no comando MQSC **DEFINE CHANNEL** ou no comando MQSC **ALTER CHANNEL** .

"Especificando que Apenas CipherSpecs Certificados por FIPS São Usados no Tempo de Execução no Cliente de MQI" na página 111
Crie seus repositórios de chaves usando software compatível com FIPS e, em seguida, especifique que o canal deve usar CipherSpecs certificados por FIPS.

"Criptografia do Conjunto B da NSA no IBM WebSphere MQ" na página 31
Este tópico fornece informações sobre como configurar IBM WebSphere MQ em sistemas Windows, Linuxe UNIX para conformidade com o perfil do TLS 1.2 compatível com o Suite B.

"Criptografia do Conjunto B da Agência Nacional de Segurança (NSA)" na página 20
O governo dos Estados Unidos da América produz aconselhamento técnico sobre sistemas e segurança de TI, inclusive criptografia de dados. A Agência Nacional de Segurança dos Estados Unidos (NSA) recomenda um conjunto de algoritmos criptográficos interoperável em seu padrão do Conjunto B.

CipherSpec valores suportados em IBM WebSphere MQ

O conjunto de CipherSpecs padrão permite somente os valores a seguir:

TLS 1.0

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA

TLS 1.2

- ECDHE_ECDSA_AES_128_CBC_SHA256
- ECDHE_ECDSA_AES_256_CBC_SHA384
- ECDHE_ECDSA_AES_128_GCM_SHA256
- ECDHE_ECDSA_AES_256_GCM_SHA384
- ECDHE_RSA_AES_128_CBC_SHA256

- ECDHE_RSA_AES_256_CBC_SHA384
- ECDHE_RSA_AES_128_GCM_SHA256
- ECDHE_RSA_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384

Ativando CipherSpecs descontinuado

Por padrão, você não tem permissão para especificar um CipherSpec descontinuado em uma definição de canal. Se você tentar especificar um CipherSpecdescontinuado, você receberá a mensagem AMQ9788 no log de erros para o gerenciador de filas

É possível reativar o CipherSpecs descontinuado editando o arquivo `qm.ini` . Na sub-rotina SSL do arquivo `qm.ini` , inclua a linha a seguir:

```
SSL:
AllowWeakCipherSpec=Yes
```

Também é possível reativar um ou mais dos CipherSpecs descontinuados no tempo de execução no servidor configurando a variável de ambiente *AMQ_SSL_WEAK_CIPHER_ENABLE* para qualquer valor. Essa variável de ambiente ativa o CipherSpecs , independentemente do valor especificado no arquivo `qm.ini` .

Registros de Autenticação de Canal

Para exercer maior controle preciso sobre o acesso concedido à conexão de sistemas em um nível de canal, é possível usar registros de autenticação de canal.

Você pode achar que os clientes tentam se conectar ao seu gerenciador de filas usando um ID do usuário em branco ou um ID do usuário de alto nível que permitiria ao cliente executar ações indesejáveis. É possível bloquear o acesso a esses clientes usando os registros de autenticação de canal. Alternativamente, um cliente pode declarar um ID do usuário que seja válido na plataforma do cliente, mas é desconhecido ou de um formato inválido na plataforma do servidor. É possível usar um registro de autenticação de canal para mapear o ID do usuário declarado para um ID do usuário válido.

Você pode achar um aplicativo cliente que se conecta ao seu gerenciador de filas e se comporta indevidamente de algum modo. Para proteger o servidor contra os problemas que este aplicativo está causando, ele precisar ser bloqueado temporariamente usando o endereço IP no qual o aplicativo cliente está até o momento em que as regras de firewall são atualizadas ou o aplicativo cliente é corrigido. É possível usar um registro de autenticação de canal para bloquear o endereço IP a partir do qual o aplicativo cliente se conecta.

Se tiver configurado uma ferramenta de administração, tal como IBM WebSphere MQ Explorer, e um canal para esse uso específico, você pode desejar assegurar que apenas computadores clientes específicos possam usá-lo. É possível usar um registro de autenticação de canal para permitir que o canal seja usado apenas a partir de determinados endereços IP.

Se você estiver apenas iniciando alguns aplicativos de amostra executando como clientes, consulte Preparando e executando os programas de amostra para obter um exemplo de como configurar o gerenciador de filas de maneira segura usando registros de autenticação de canal.

Para obter registros de autenticação de canal para controlar canais de entrada, use o comando MQSC **`ALTER QMGR CHLAUTH(ENABLED)`**.

As regras de **CHLAUTH** são aplicadas para um canal MCA que é criado em resposta a uma nova conexão de entrada. Para um canal MCA criado em resposta ao canal sendo iniciado localmente, nenhuma regra de **CHLAUTH** é aplicada.

Tabela 4. Quando as regras de CHLAUTH são aplicadas para pares de canais diferentes	
Tipo de canal	**MCA no qual as regras CHLAUTH são aplicadas**
SDR-RCVR	RCVR
RQSTR-SVR (iniciado em SVR)	RQSTR
RQSTR-SVR (iniciado em RQSTR)	SVR
RQSTR-SDR (iniciado em SDR)	RQSTR
RQSTR-SDR (iniciado em RQSTR)	SDR para conexão inicial. RQSTR para conexão de retorno de chamada.

Os registros de autenticação de canal podem ser criados para executar as seguintes funções:

- Bloquear as conexões dos endereços IP específicos.
- Bloquear as conexões de IDs de usuário específicos.
- Configurar um valor MCAUSER a ser usado para qualquer canal que se conecta a partir de um endereço IP específico.
- Configurar um valor MCAUSER a ser usado para qualquer canal declarando um ID do usuário específico.
- Configurar um valor MCAUSER a ser usado para qualquer canal que tenha um SSL específico ou Nome Distinto TLS (DN).
- Configurar um valor MCAUSER a ser usado para qualquer canal que se conecta a partir de um gerenciador de filas específico.
- Bloquear as conexões consideradas de um certo gerenciador de filas, a menos que a conexão seja de um endereço IP específico.
- Bloquear conexões que apresentam um certo certificado SSL ou TLS, a menos que a conexão seja de um endereço IP específico.

Estas utilizações são explicadas em detalhes nas seções a seguir.

Você cria, modifica ou remove registros de autenticação de canais usando o comando MQSC **SET CHLAUTH** ou o comando PCF **Set Channel Authentication Record**.

Nota: Grandes números de registros de autenticação de canal podem ter um impacto negativo no desempenho de um gerenciador de filas.

Bloqueando endereços IP

Normalmente é a função de um firewall evitar o acesso de determinados endereços IP. No entanto, podem existir ocasiões nas quais você sofre tentativas de conexão de um endereço IP que não deveria ter acesso ao seu sistema do WebSphere MQ e deve bloquear temporariamente o endereço antes do firewall poder ser atualizado. Essas tentativas de conexão podem nem estar vindo dos canais do WebSphere MQ , mas de outros aplicativos de soquete que estão mal configurados para destinar o listener do WebSphere MQ . Bloqueie os endereços IP configurando um registro de autenticação de canal do tipo BLOCKADDR. É possível especificar um ou mais endereços únicos, intervalos de endereços ou padrões, incluindo curingas.

Sempre que uma conexão de entrada é recusada porque o endereço IP está bloqueado desta maneira, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_ADDRESS é emitida, desde que os eventos do canal estejam ativados e o gerenciador de filas esteja em execução. Além disso, a conexão é mantida aberta por 30 segundos antes de retornar o erro para assegurar que o listener não estoure com tentativas repetidas de conexão que estão bloqueadas.

Para bloquear endereços IP somente em canais específicos ou para evitar o atraso antes do erro ser relatado, configure um registro de autenticação de canal do tipo ADDRESSMAP com o parâmetro USERSRC(NOACCESS).

Sempre que uma conexão de entrada é recusada por esta razão, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_NOACCESS é emitida, desde que os eventos do canal estejam ativados e o gerenciador de filas esteja em execução.

Consulte “Bloqueando Endereços IP Específicos” na página 185 para obter um exemplo.

Bloqueando IDs de usuário

Para evitar que determinados IDs do usuário se conectem por meio de um canal do cliente, configure o registro de autenticação de canal do tipo BLOCKUSER. Este tipo de registro de autenticação de canal se aplica somente a canais do cliente, não a canais de mensagens. É possível especificar um ou mais IDs de usuários individuais a serem bloqueados, mas você não pode usar curingas.

Sempre que uma conexão de entrada é recusada por esta razão, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_USERID é emitida, desde que os eventos do canal estejam ativados.

Consulte “Bloqueando IDs de Usuários Específicos” na página 187 para obter um exemplo.

Também é possível bloco qualquer acesso para IDs de usuários especificados em determinados canais configurando um registro de autenticação de canal do tipo USERMAP com o parâmetro USERSRC(NOACCESS).

Sempre que uma conexão de entrada é recusada por esta razão, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_NOACCESS é emitida, desde que os eventos do canal estejam ativados e o gerenciador de filas esteja em execução.

Consulte “Bloqueando Acesso para um ID do Usuário Declarado do Cliente” na página 190 para obter um exemplo.

Bloqueando nomes do gerenciador de filas

Para especificar que qualquer canal que se conecta a partir de um gerenciador de filas especificado não deve ter acesso, configure um registro de autenticação de canal do tipo QMGRMAP com o parâmetro USERSRC(NOACCESS). É possível especificar um único nome do gerenciador de filas ou um padrão incluindo curingas. Não há equivalente da função BLOCKUSER para bloquear o acesso de gerenciadores de filas.

Sempre que uma conexão de entrada é recusada por esta razão, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_NOACCESS é emitida, desde que os eventos do canal estejam ativados e o gerenciador de filas esteja em execução.

Consulte “Bloqueando Acesso de um Gerenciador de Filas Remotas” na página 189 para obter um exemplo.

Bloqueando DNs de SSL ou TLS

Para especificar que qualquer usuário que apresenta um certificado pessoal de SSL ou TLS contendo um DN especificado não possui acesso, configure um registro de autenticação de canal do tipo SSLPEERMAP com o parâmetro USERSRC(NOACCESS). É possível especificar um único nome distinto ou um padrão incluindo curingas. Não há equivalente da função BLOCKUSER para bloquear o acesso para DNs.

Sempre que uma conexão de entrada é recusada por esta razão, uma mensagem do evento MQRC_CHANNEL_BLOCKED com o qualificador de razão MQRQ_CHANNEL_BLOCKED_NOACCESS é emitida, desde que os eventos do canal estejam ativados e o gerenciador de filas esteja em execução.

Consulte “Bloqueando Acesso para um Nome Distinto de SSL” na página 190 para obter um exemplo.

Mapeando endereços IP para IDs do usuário para serem usados

Para especificar se algum canal conectando-se a partir de um endereço IP especificado deve usar um MCAUSER específico, configure um registro de autenticação de canal do tipo ADDRESSMAP. É possível especificar um único endereço, um intervalo de endereços ou um padrão incluindo curingas.

Se você usar um encaminhador de porta, quebra da sessão DMZ ou qualquer outra configuração que altera o endereço IP apresentado ao gerenciador de filas, o mapeamento de endereços IP não é necessariamente adequado para uso.

Consulte “Mapeando um Endereço IP para um ID do Usuário MCAUSER” na página 190 para obter um exemplo.

Mapeando nomes do gerenciador de filas para IDs do usuário para serem usados

Para especificar se algum canal conectando-se a partir de um gerenciador de filas especificado deve usar um MCAUSER específico, configure um registro de autenticação de canal do tipo QMGRMAP. É possível especificar um único nome do gerenciador de filas ou um padrão incluindo curingas.

Consulte “Mapeando um Gerenciador de Filas Remoto para um ID do Usuário MCAUSER” na página 187 para obter um exemplo.

Mapeando IDs de Usuários Declarados por um Cliente para IDs de Usuários a Serem Usados

Para especificar que, quando um determinado ID do usuário é usado por uma conexão a partir de um cliente do WebSphere MQ MQI, um MCAUSER diferente especificado deve ser usado, configure um registro de autenticação de canal do tipo USERMAP. O mapeamento do ID do usuário não usa curingas.

Consulte “Mapeando um ID do Usuário Decrarado pelo Cliente para um ID do Usuário MCAUSER” na página 188 para obter um exemplo.

Mapeando DNs SSL ou TLS para IDs do usuário para serem usados

Para especificar se algum usuário apresentando um certificado pessoal de Secure Sockets Layer/ Segurança da Camada de Transporte contendo um Nome distinto especificado deve usar um MCAUSER específico, configure um registro de autenticação de canal do tipo SSLPEERMAP. É possível especificar um único nome distinto ou um padrão incluindo curingas.

Consulte “Mapeando um Nome Distinto de SSL ou TLS para um ID do Usuário MCAUSER” na página 189 para obter um exemplo.

Mapeamento de Gerenciadores de Filas, Clientes ou DNs de SSL ou TLS de acordo com Endereço IP

Em algumas circunstâncias, pode ser possível para um terceiro imitar um nome do gerenciador de filas. Um certificado SSL ou TLS ou arquivo do banco de dados de chave também pode ser deturpado e reutilizado. Para se proteger contra essas ameaças, é possível especificar que uma conexão a partir de um determinado gerenciador de filas ou cliente, ou usando um determinado Nome distinto deve ser estabelecida a partir de um endereço IP especificado. Configure um registro de autenticação de canal do tipo USERMAP, QMGRMAP ou SSLPEERMAP e especifique o endereço IP permitido, ou padrão de endereços IP, usando o parâmetro ADDRESS.

Consulte “Mapeando um Gerenciador de Filas Remoto para um ID do Usuário MCAUSER” na página 187 para obter um exemplo.

Interação entre registros de autenticação de canal

É possível que um canal ao tentar fazer uma conexão corresponda a mais de um registro de autenticação de canal e que isso tenha efeitos contraditórios. Por exemplo, um canal pode declarar um ID do usuário que foi bloqueado por um registro de autenticação de canal BLOCKUSER, mas com um certificado SSL ou TLS que corresponde a um registro SSLPEERMAP que configura um ID do usuário diferente. Além disso, se registros de autenticação de canal usarem curingas, um único endereço IP, nome do gerenciador de filas ou Nome distinto Secure Sockets Layer ou de Segurança da Camada de Transporte pode corresponder a vários padrões. Por exemplo, o endereço IP 192.0.2.6 corresponde aos padrões 192.0.2.0-24, 192.0.2. *, e 192.0.*.6. A ação executada é determinada conforme a seguir.

- O registro de autenticação de canal usado é selecionado conforme a seguir:

  - Um registro de autenticação de canal que corresponde explicitamente ao nome de canal tem prioridade sobre um registro de autenticação de canal que corresponde ao nome de canal usando um curinga.
  - Um registro de autenticação de canal usando um DN SSL ou TLS tem prioridade sobre um registro que usa um ID do usuário, nome do gerenciador de filas ou endereço IP.
  - Um registro de autenticação de canal que usa um ID do usuário ou um nome do gerenciador de filas tem prioridade sobre um registro que usa um endereço IP.
- Se um registro de autenticação de canal correspondente for localizado e ele especificar um MCAUSER, este MCAUSER será designado ao canal.
- Se um registro de autenticação de canal correspondente for localizado e ele especificar que o canal não possui acesso, um valor de MCAUSER igual a *NOACCESS será designado ao canal. Este valor pode, posteriormente, ser alterado por um programa de saída de segurança.
- Se nenhum registro de autenticação de canal correspondente for localizado, ou um registro de autenticação de canal correspondente for localizado e ele especificar que o ID do usuário do canal deve ser usado, o campo MCAUSER será inspecionado.
  - Se o campo MCAUSER estiver em branco, o ID do usuário do cliente é designado ao canal.
  - Se o campo MCAUSER não estiver em branco, ele será designado ao canal.
- Qualquer programa de saída de segurança é executado. Este programa de saída pode configurar o ID do usuário do canal ou determinar que o acesso deve ser bloqueado.
- Se a conexão estiver bloqueada ou o MCAUSER estiver configurado para *NOACCESS, o canal é encerrado.
- Se a conexão não estiver bloqueada, para qualquer canal exceto um canal do cliente, o ID do usuário do canal determinado nas etapas anteriores será verificado com relação à lista de usuários bloqueados.
  - Se o ID do usuário estiver na lista de usuários bloqueados, o canal será encerrado.
  - Se o ID do usuário não estiver na lista de usuários bloqueados, o canal é executado.

Onde um número de registros de autenticação de canal corresponde com um nome de canal, endereço IP, nome do gerenciador de filas ou Nome distinto Secure Sockets Layer ou de Segurança da Camada de Transporte, a correspondência mais específica será usada. A correspondência considerada mais específico é determinada conforme a seguir.

- Para um nome de canal:
  - A correspondência mais específica é um nome sem caracteres curinga, por exemplo A.B.C.
  - A correspondência mais genérica é um único asterisco (*), que corresponde a todos os nomes de canal.
  - Um padrão com um asterisco na posição mais à esquerda é mais genérico que um padrão com um valor definido na posição mais à esquerda. Assim, *.B.C é mais genérico do que A.*.
  - Um padrão com um asterisco na segunda posição é mais genérico que um padrão com um valor definido na segunda posição, e semelhantemente para cada posição subsequente. Assim, A.*.C é mais genérico do que A.B.*
  - Onde dois ou mais padrões possuem um asterisco na mesma posição, o com menos nós após o asterisco é mais genérico. Assim A. * é mais genérico do que A.*.C
- Para um endereço IP:
  - A correspondência mais específica é um nome sem curingas, por exemplo 192.0.2.6.
  - A correspondência mais genérica é um único asterisco (*), que corresponde a todos os nomes de canal.
  - Um padrão com um asterisco na posição mais à esquerda é mais genérico que um padrão com um valor definido na posição mais à esquerda. Assim, *.0.2.6 é mais genérico do que 192.*.
  - Um padrão com um asterisco na segunda posição é mais genérico que um padrão com um valor definido na segunda posição, e semelhantemente para cada posição subsequente. Assim, 192.*.2.6 é mais genérico do que 192.0.*.

  - Onde dois ou mais padrões possuem um asterisco na mesma posição, o com menos nós após o asterisco é mais genérico. Assim, 192 .* é mais genérico do que 192.*.2.*.
  - Um intervalo indicado com um hífen (-) é mais específico do que com um asterisco. Portanto, 192.0.2.0-24 é mais específico do que 192.0.2.*.
  - Um intervalo que é um subconjunto de um outro é mais específico do que o intervalo maior. Portanto, 192.0.2.5-15 é mais específico do que 192.0.2.0-24.
  - A sobreposição de intervalos não é permitida. Por exemplo, não é possível ter registros de autenticação de canal para 192.0.2.0-15 e 192.0.2.10-20.
  - Um padrão não pode ter menos do que o número necessário de partes, a menos que o padrão termine com um único asterisco final. Por exemplo 192.0.2 é inválido, mas 192.0.2 .* é válido
  - Um asterisco final deve ser separado do restante do endereço pelo separador de parte apropriado (um ponto (.) para IPv4, dois pontos (:) para IPv6). Por exemplo, 192.0* não é válido porque o asterisco não está em uma parte própria sua.
  - Um padrão pode conter asteriscos adicionais desde que nenhum asterisco seja adjacente ao asterisco final. Por exemplo, 192.*.2.* é válido, mas 192.0.*.* não é válido.
  - Um padrão de endereço do IPv6 não pode conter dois pontos duplos e um asterisco final, pois o endereço resultante ficaria ambíguo. Por exemplo, 2001::* poderia expandir para 2001:0000:*, 2001:0000:0000:* e assim por diante.
- Para um nome do gerenciador de filas:
  - A correspondência mais específica é um nome sem curingas, por exemplo 192.0.2.6.
  - A correspondência mais genérica é um único asterisco (*), que corresponde a todos os nomes de canal.
  - Um padrão com um asterisco na posição mais à esquerda é mais genérico que um padrão com um valor definido na posição mais à esquerda. Assim, *QUEUEMANAGER é mais genérico do que QUEUEMANAGER*.
  - Um padrão com um asterisco na segunda posição é mais genérico que um padrão com um valor definido na segunda posição, e semelhantemente para cada posição subsequente. Assim, Q*MANAGER é mais genérico do que QUEUE*.
  - Onde dois ou mais padrões possuem um asterisco na mesma posição, o com menos caracteres após o asterisco é mais genérico. Assim, Q* é mais genérico do que Q*MGR.
- Para um Nome Distinto (DN) SSL ou TLS, a ordem de precedência das subsequências é a seguinte:

Tabela 5. Ordem de precedência de subsequências

Ordem	**Subsequência do DN**	**Nome**
1	SERIALNUMBER=	Número de série do certificado
2	MAIL=	Endereço eletrônico
3	E=	Endereço de e-mail (descontinuado na preferência para MAIL)
4	UID=, USERID=	Identificador de usuário
5	CN=	Common name
6	T =	Título
7	OU=	unidade organizacional
8	DC=	Componente de domínio
9	O=	Organização
10	STREET=	Rua / Primeira linha do endereço

Tabela 5. Ordem de precedência de subsequências (continuação)		
Ordem	**Subsequência do DN**	**Nome**
11	L=	Localidade
12	ST=, SP=, S=	Nome do estado ou território
13	PC=	Código Postal / Código de Endereçamento Postal
14	C=	País
15	UNSTRUCTUREDNAME=	Nome do host
16	UNSTRUCTUREDADDRESS=	Endereço IP
17	DNQ=	Qualificador de Nome Distinto

Assim, se um certificado de Secure Sockets Layer ou de Segurança da Camada de Transporte for apresentado com um Nome distinto contendo as subsequências O=IBM e C=UK, o WebSphere MQ usará um registro de autenticação de canal para O=IBM em detrimento de um para C=UK, se ambos estiverem presentes.

Um Nome distinto pode conter diversas OUs, que devem ser especificadas em ordem hierárquica com as maiores unidades organizacionais especificadas primeiro. Se dois Nomes distintos forem iguais em todos os aspectos, exceto por seus valores de OU, o Nome distinto mais específico será determinado conforme a seguir:

1. Se eles possuírem números diferentes de atributos de OU, o Nome distinto com a maioria dos valores de OU é mais específico. Isso porque o Nome distinto com mais Unidades Organizacionais qualifica integralmente o Nome distinto em mais detalhes e fornece mais critérios de correspondência. Mesmo se sua OU de nível superior for um curinga (OU=*), o DN com mais OUs ainda será considerado o mais específico no geral.
2. Se eles tiverem o mesmo número de atributos de OU, os pares correspondentes de valores de OU são comparados na sequência da esquerda-para-direita, em que a OU mais à esquerda é o nível mais superior (menos específico), de acordo com as seguintes regras.
   a. Uma OU sem nenhum valor curinga é a mais específica porque ela pode corresponder exatamente com uma sequência apenas.
   b. Uma OU com um único curinga no início ou no final (por exemplo, OU=ABC* ou OU=*ABC) é a próxima mais específica.
   c. Uma OU com dois curingas (por exemplo, OU=*ABC*) é a próxima mais específica.
   d. Uma OU que consiste em somente um asterisco (OU=*) é a menos específica.
3. Se a comparação de sequência tiver uma ligação entre dois valores de atributo com a mesma especificidade, a sequência de atributos mais longa será mais específica.
4. Se a comparação de sequência estiver empatada entre dois valores de atributo de mesma especificidade e comprimento, então o resultado será determinado por uma comparação de sequência sem distinção entre maiúsculas e minúsculas da parte do Nome distinto, excluindo quaisquer curingas.

Se dois DNs forem iguais em todos os aspectos, exceto por seus valores de DC, as mesmas regras de correspondência se aplicarão para OUs, exceto que em valores de DC, o DC mais à esquerda é o nível mais baixo (mais específico) e a ordenação de comparação difere em conformidade.

Exibindo registros de autenticação de canal

Para exibir registros de autenticação de canal, use o comando MQSC **DISPLAY CHLAUTH** ou o comando PCF **Inquire Channel Authentication Records**. É possível escolher para retornar todos os registros que correspondam ao nome de canal fornecido ou é possível escolher um correspondência explícita. A correspondência explícita informa qual registro de autenticação canal seria usado se um canal

tentasse fazer uma conexão a partir de um endereço IP específico, a partir de um gerenciador de filas específico ou usando um ID do usuário específico e, opcionalmente, apresentando um certificado pessoal de Secure Sockets Layer/Segurança da Camada de Transporte contendo um Nome distinto especificado.

Conceitos relacionados

"Segurança para o Sistema de Mensagens Remoto" na página 57
Esta seção trata dos aspectos do sistema de mensagens remoto de segurança.

Segurança de mensagem em IBM WebSphere MQ ..

A segurança de mensagem na infraestrutura do IBM WebSphere MQ é fornecida por um componente licenciado separadamente IBM WebSphere MQ Advanced Message Security.

IBM WebSphere MQ Advanced Message Security (AMS) expande os serviços de segurança do IBM WebSphere MQ para fornecer assinatura e criptografia de dados no nível da mensagem. Os serviços expandidos garantem que os dados da mensagem não foram modificados entre quando foram originalmente colocados em uma fila e quando foram recuperados. Além disso, o AMS verifica se um emissor de dados da mensagem está autorizado a colocar mensagens assinadas em uma fila de destino.

Conceitos relacionados

"IBM WebSphere MQ Advanced Message Security" na página 274
IBM WebSphere MQ Advanced Message Security (AMS) é um componente licenciado separadamente do IBM WebSphere MQ Advanced Message Security que fornece um alto nível de proteção para dados sensíveis que fluem pela rede do IBM WebSphere MQ Advanced Message Security , enquanto não impacta os aplicativos finais.

Planejando para seus requisitos de segurança

Esta coleção de tópicos explica o que é necessário considerar ao planejar a segurança em um ambiente do IBM WebSphere MQ.

É possível usar o IBM WebSphere MQ para uma grande variedade de aplicativos em uma gama de plataformas. Os requisitos de segurança provavelmente serão diferentes para cada aplicativo. Para alguns, a segurança será uma consideração crítica.

O WebSphere MQ fornece uma variedade de serviços de segurança de nível de link, incluindo suporte para o Secure Sockets Layer (SSL) e o Transport Layer Security (TLS).

Você deve considerar determinados aspectos de segurança ao implementar o WebSphere. Nos sistemas UNIX, Linux e Windows , se você ignorar esses aspectos e não fazer nada não poderá usar o WebSphere MQ.

Considerações de segurança são descritas abaixo.

Autoridade para administrar o WebSphere MQ

WebSphere MQ administradores precisam de autoridade para:

- Emita comandos para administrar o WebSphere MQ
- Use o IBM WebSphere MQ Explorer

Para obter informações adicionais, consulte:

- "Autoridade para administrar IBM WebSphere MQ em sistemas UNIX, Linux, and Windows" na página 200

Autoridade para trabalhar com objetos do WebSphere MQ

Os aplicativos podem acessar os seguintes objetos do WebSphere MQ emitindo chamadas MQI:

- Gerenciadores de filas
- Filas

- Processos
- Listas de Nomes
- tópicos

Os aplicativos também podem usar comandos Programmable Command Format (PCF) para acessar esses objetos do WebSphere MQ e também acessar canais e objetos de informações sobre autenticação. Esses objetos podem ser protegidos pelo WebSphere MQ para que os IDs de usuário associados aos aplicativos precisem de autoridade para acessá-los.

Para obter mais informações, consulte “Autorização para aplicativos usarem o IBM WebSphere MQ” na página 51.

Segurança de canal

Os IDs de usuário associados aos agentes do canal de mensagens (MCAs) precisam de autoridade para acessar vários recursos do WebSphere MQ ... Por exemplo, um MCA precisa ser capaz de conectar-se a um gerenciador de filas. Se estiver enviando MCA, deverá estar apto a abrir a fila de transmissão do canal. Se for um MCA de recepção, precisa ser capaz de abrir as filas de destino. Os IDs de usuário associados aos aplicativos que precisam administrar canais, inicializadores de canais e listeners precisam de autoridade para usar os comandos do PCF relevantes. No entanto, a maioria dos aplicativos não precisa desse acesso.

Para obter mais informações, consulte “Autorização de canal” na página 71.

Considerações Adicionais

Será necessário considerar os aspectos de segurança a seguir somente se você estiver usando determinadas funções do WebSphere MQ ou extensões do produto base:

- “Segurança para Clusters de Gerenciadores de Filas” na página 80
- “Segurança para Publicação / Assinatura do IBM WebSphere MQ” na página 81
- “Segurança para o intermediário de Internet do IBM WebSphere MQ” na página 83

Planejando a identificação e a autenticação

Decida quais IDs do usuário usar e como e em que níveis você deseja aplicar controles de autenticação.

Deve-se como você identificará os usuários dos seus aplicativos IBM WebSphere MQ, tendo em conta que os sistemas operacionais suportam diferentes IDs de usuário de comprimentos diferentes. É possível usar registros de autenticação de canal para mapear de um ID do usuário para outro, ou para especificar um ID do usuário com base em alguns atributos da conexão. Os canais do IBM WebSphere MQ que usam SSL ou TLS usam certificados digitais como um mecanismo para identificação e autenticação. Cada certificado digital tem um nome distinto do assunto que pode ser mapeado para identidades específicas usando registros de autenticação de canal. Além disso, os certificados de autoridade de certificação no repositório de chaves determinam quais certificados digitais podem ser usados para autenticar para o IBM WebSphere MQ. Para obter informações adicionais, consulte:

- “Mapeando um Gerenciador de Filas Remoto para um ID do Usuário MCAUSER” na página 187
- “Mapeando um ID do Usuário Decrarado pelo Cliente para um ID do Usuário MCAUSER” na página 188
- “Mapeando um Nome Distinto de SSL ou TLS para um ID do Usuário MCAUSER” na página 189
- “Mapeando um Endereço IP para um ID do Usuário MCAUSER” na página 190

Planejando a Autenticação para um Aplicativo Cliente

É possível aplicar controles de autenticação em quatro níveis: no nível de comunicações, em saídas de segurança, com registros de canal de autenticação, e em termos de identificação que é transmitida para uma saída de segurança.

Há quatro níveis de segurança a serem considerados. O diagrama mostra um cliente MQI IBM WebSphere MQ que está conectado a um servidor. A segurança é aplicada em quatro níveis, conforme descrito no texto a seguir. MCA é um Agente do Canal de Mensagem.

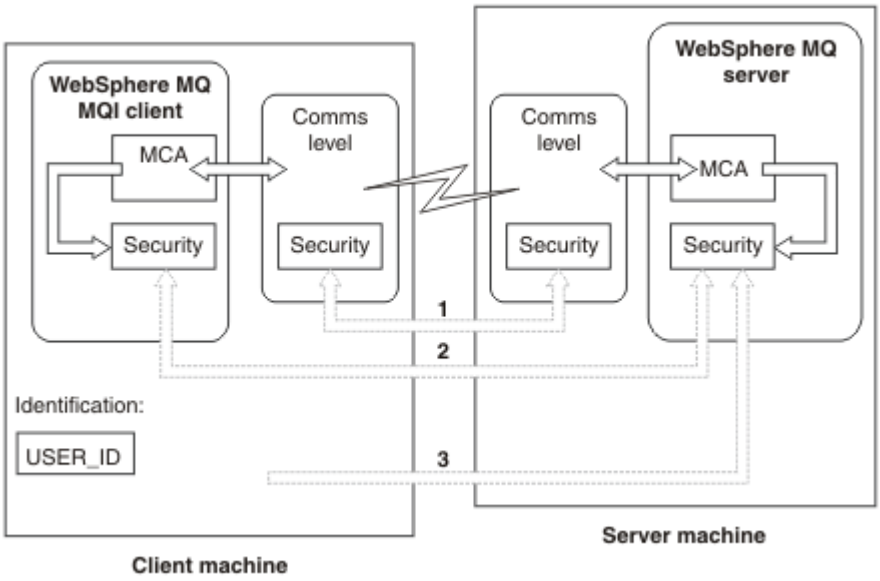


Figura 7. Segurança em uma Conexão de Cliente/Servidor

1. Nível de comunicações

   Veja a seta 1. Para implementar a segurança no nível de comunicações, use SSL ou TLS Para obter informações adicionais, consulte “Protocolos de Segurança Criptográficos: SSL e TLS” na página 14

2. Registros de Autenticação de Canal

   Ver as setas 2 e 3. A autenticação pode ser controlada usando o endereço IP ou nomes distintos SSL/TLS no nível de segurança. Um ID do usuário também pode ser bloqueado ou um ID do usuário declarado pode ser mapeado para um ID de usuário válido. Uma descrição completa é fornecida em “Registros de Autenticação de Canal” na página 40.

3. Saídas de Segurança do Canal

   Veja a seta 2. As saídas de segurança do canal para a comunicação do cliente para o servidor podem funcionar da mesma forma que para a comunicação do servidor para servidor. Um par de protocolos independentes de saída podem ser gravados para fornecer autenticação mútua entre o cliente e o servidor. Uma descrição completa é fornecida em Programas de saída de segurança de canal.

4. Identificação que é transmitida a uma saída de segurança do canal

   Veja a seta 3. Na comunicação do cliente para o servidor, as saídas de segurança do canal não têm que operar como um par. A saída no IBM WebSphere MQ ao lado do cliente podem ser omitidos. Neste caso, o ID do usuário é colocado no descritor de canal (MQCD) e a saída de segurança do lado do servidor pode alterá-lo, se necessário.

   Os clientes do Windows também enviam informações adicionais para ajudar na identificação.

   - O ID do usuário que é transmitido ao servidor é o ID do usuário com login efetuado atualmente no cliente.
   - O ID de segurança do usuário com logon efetuado atualmente.

   Para ajudar na identificação do cliente IBM WebSphere MQ para HP Integrity NonStop Server, o cliente transmite o OSS Defenda o alias sob qual aplicativo do cliente está em execução. Esse ID tem geralmente o formato <PRIMARYGROUP>.<ALIAS>. Se necessário, você pode mapear esse ID do usuário para um ID de usuário alternativo no gerenciador de filas usando a registros de autenticação de canal ou uma saída de segurança. Para obter mais informações sobre saídas de mensagem, consulte “Mapeamento de identidade em saídas de mensagem” na página 151. Para obter

informações adicionais sobre a definição de registros de autenticação de canal, consulte “Mapeando um ID do Usuário Decrarado pelo Cliente para um ID do Usuário MCAUSER” na página 188.

Os valores do ID do usuário e, se disponíveis, o ID de segurança, podem ser usados pela saída de segurança do servidor para estabelecer a identidade do cliente MQI IBM WebSphere MQ .

IDs de Usuário

Se o cliente MQI IBM WebSphere MQ estiver no Windows e o servidor IBM WebSphere MQ também estiver no Windows e tiver acesso ao domínio no qual o ID do usuário cliente está definido, o IBM WebSphere MQ suportará IDs do usuário de até 20 caracteres. Nas plataformas e configurações do UNIX and Linux, o comprimento máximo é de 12 caracteres.

Um servidor WebSphere MQ para Windows não suporta a conexão de um cliente Windows se o cliente estiver em execução sob um ID do usuário que contém o caractere @, por exemplo `abc@d`. O código de retorno para a chamada `MQCONN` no cliente é MQRC_NOT_AUTHORIZED.

No entanto, é possível especificar o ID do usuário usando dois caracteres @, por exemplo, `abc@@d`. Usar o formato `id@domain` é a prática preferencial para assegurar que o ID do usuário seja resolvido no domínio correto de forma consistente; portanto, `abc@@d@domain`.

Observe que `UNKNOWN` é um ID do usuário reservado e o ID do usuário `NOBODY` também tem significados especiais para WebSphere MQ. Criar IDs de usuário no sistema operacional chamados `UNKNOWN` ou `NOBODY` pode ter resultados indesejados.

Embora os IDs de usuário sejam usados para autenticar, os grupos são usados para autorização, exceto para o Windows

Se você criar contas de serviço, sem prestar atenção em grupos, e autorizar todos os IDs de usuário diferentes, cada usuário poderá acessar as informações dos outros usuários.

Planejando a autorização

Planeje os usuários que terão autoridade administrativa e planeje como autorizar usuários de aplicativos a usarem adequadamente os objetos do IBM WebSphere MQ , incluindo aqueles que se conectam a partir de um cliente MQI do IBM WebSphere MQ .

Deve ser concedido acesso a indivíduos ou aplicações para usar o IBM WebSphere MQ. O acesso que eles requerem depende das funções que eles realizam e das tarefas que eles precisam executar. A autorização no IBM WebSphere MQ pode ser subdividida em duas categorias principais:

- Autorização para executar operações administrativas
- Autorização para aplicativos usarem o IBM WebSphere MQ

As classes de operação são controladas pelo mesmo componente, e a um individual pode ser concedida a autoridade para executar ambas as categorias de operação.

Os tópicos a seguir fornecem informações adicionais sobre áreas específicas de autorização que deve ser consideradas:

Autoridade para administrar o IBM WebSphere MQ

Os administradores do IBM WebSphere MQ precisam de autoridade para executar várias funções. Essa autoridade é obtida de diferentes maneiras em diferentes plataformas.

Administradores do IBM WebSphere MQ precisam de autoridade para:

- Emita comandos para administrar o IBM WebSphere MQ
- Use oIBM WebSphere MQ Explorer

Para obter mais informações, consulte o tópico apropriado para seu sistema operacional.

Autoridade para administrar IBM WebSphere MQ em sistemas UNIX e Windows

Um administrador do IBM WebSphere MQ é um membro do grupo mqm. Este grupo tem acesso a todos os recursos do IBM WebSphere MQ e pode emitir comandos de controle do IBM WebSphere MQ. Um administrador pode conceder autoridades específicas para outros usuários.

Para ser um administrador IBM WebSphere MQ nos sistemas UNIX e Windows , um usuário deve ser um membro do *grupo mqm*. Esse grupo é criado automaticamente quando você instala o WebSphere MQ Para permitir que os usuários emitam comandos de controle, deve-se incluí-los no grupo mqm. Isso inclui o usuário raiz em sistemas UNIX

Os usuários que não são membros do grupo mqm podem ser receber privilégios administrativos, mas eles não são capazes de emitir comandos de controle do IBM WebSphere MQ e estão autorizados a executar apenas os comandos para os quais tiverem recebido acesso.

Além disso, em sistemas Windows , as contas SYSTEM e de Administrador têm acesso total aos recursos do IBM WebSphere MQ

Todos os membros do grupo mqm têm acesso a todos os recursos do WebSphere MQ no sistema, incluindo a possibilidade de administrar qualquer gerenciador de filas em execução no sistema. Esse acesso pode ser revogado somente com a remoção de um usuário do grupo mqm. Nos sistemas Windows , os membros do grupo de Administradores também têm acesso a todos os recursos do WebSphere MQ

Os administradores podem usar o comando de controle **runmqsc** para emitir comandos WebSphere MQ Script (MQSC). Quando **runmqsc** é utilizado no modo indireto para enviar comandos MQSC para um gerenciador de fila remoto, cada comando será encapsulado dentro de um comando PCF Escape. Os administradores devem ter as autoridades requeridas para os comandos MQSC serem processados pelo gerenciador de fila remoto.

O WebSphere MQ Explorer emite comandos PCF para executar tarefas de administração. Os administradores não requerem nenhuma autoridade adicional para usar o WebSphere MQ Explorer para administrar um gerenciador de filas no sistema local. Quando o WebSphere MQ Explorer é usado para administrar um gerenciador de fila em outro sistema, os administradores devem ter as autoridades necessárias para que os comandos PCF sejam processados pelo gerenciador de filas remotas

Para obter mais informações sobre as verificações de autoridade efetuadas quando os comando de PCF e MQSC são processados, consulte os seguintes tópicos:

- Para comandos que operam em gerenciadores de filas, filas, canais, processos, listas de nomes e objetos de informações sobre autenticação, consulte “Autorização para aplicativos usarem o IBM WebSphere MQ” na página 51.
- Para comandos que operam em canais, inicializadores de canais, listeners e clusters, consulte Segurança de canal.

Para obter mais informações sobre a autoridade necessária para administrar o WebSphere MQ em sistemas UNIX e Windows , consulte as informações relacionadas.

Autorização para aplicativos usarem o IBM WebSphere MQ

Quando os aplicativos acessam objetos, os IDs de usuário associados aos aplicativos precisam de autoridade apropriada.

Os aplicativos podem acessar os seguintes objetos do IBM WebSphere MQ, emitindo chamadas de MQI:

- Gerenciadores de filas
- Filas
- Processos
- Listas de Nomes
- tópicos

Os aplicativos também podem usar comando de PCF para administrar objetos do IBM WebSphere MQ. Quando o comando PCF é processado, ele usa o contexto de autoridade do ID do usuário que envia a mensagem de PCF.

Os aplicativos, nesse contexto, incluem aqueles escritos por usuários e fornecedores

Aplicativos que usam classes IBM WebSphere MQ para Java, classes IBM WebSphere MQ para JMS, classes IBM WebSphere MQ para .NET ou Message Service Clients for C/C++ and .NET usam o MQI indiretamente.

Os MCAs também emitem chamadas MQI e os IDs do usuário associados aos MCAs precisam de autoridade para acessar esses objetos do WebSphere MQ Para obter mais informações sobre esses IDs do usuário e as autoridades que eles requerem, consulte a seção "Autorização de canal" na página 71.

Quando as Verificações de Autoridade são Executadas

As verificações de autoridade são executadas quando um aplicativo tenta acessar um gerenciador de filas, uma fila, um processo ou uma lista de nomes.

As verificações são executadas nas seguintes situações:

Quando um aplicativo estabelece conexão com um gerenciador de fila usando uma chamada MQCONN ou MQCONNX

O gerenciador de fila solicita o sistema operacional do ID do usuário associado ao aplicativo. Então, verifica se o ID do usuário tem autorização para estabelecer conexão com ele e o retém para verificações futuras.

Os usuários não têm que efetuar sign on para o IBM WebSphere MQ. O IBM WebSphere MQ assume que os usuários estão conectados ao sistema operacional subjacente e que foram autenticados por ele.

Quando um aplicativo abre um objeto do IBM WebSphere MQ usando uma chamada MQOPEN ou MQPUT1

Todas as verificações de autoridade são executadas quando um objeto é aberto, não ao ser acessado posteriormente. Por exemplo, as verificações de autoridade são executadas quando um aplicativo abre uma fila. Elas não são executadas quando o aplicativo coloca mensagens na fila ou recebe mensagens da fila.

Quando um aplicativo abre um objeto, ele especifica os tipos de operação de que necessita executar nele. Por exemplo, um aplicativo pode abrir uma fila para consultar e obter mensagens, mas não para colocar mensagens nela. Para cada tipo de operação, o gerenciador de fila verifica se o ID do usuário associado ao aplicativo tem a autoridade para executar essa operação.

Quando um aplicativo abre uma fila, as verificações de autoridade são executadas no objeto nomeado no campo `ObjectName` do descritor de objeto. O campo `ObjectName` é usado nas chamadas `MQOPEN` ou `MQPUT1`. Se o objeto for uma fila de alias ou uma definição de fila remota, as verificações de autoridade serão executadas no próprio objeto. Elas não são executadas na fila para a qual a fila de alias ou a definição de fila remota é resolvida. Isso significa que o usuário não precisa de permissão para acessá-la. Limite a autoridade para criar filas a usuários privilegiados. Se você não fizer isto, os usuários podem efetuar bypass do controle de acesso normal simplesmente criando um alias.

Um aplicativo pode referenciar uma fila remota explicitamente. Ele configura os campos `ObjectName` e `ObjectQMgrName` no descritor de objetos para os nomes da fila remota e o gerenciador de filas remotas. As verificações de autoridade serão executadas na fila de transmissão com o mesmo nome que o gerenciador de filas remotas. No UNIX, Linux, and Windows, uma verificação será feita com relação ao perfil RQMNAME que corresponde ao nome do gerenciador de filas remotas, se o armazenamento em cluster estiver sendo utilizado Um aplicativo pode fazer referência a uma fila de clusters explicitamente, definindo o campo `ObjectName` no descritor de objeto com o nome da fila de clusters. As verificações de autoridade são executadas na fila de transmissão do cluster, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

A autoridade para uma fila dinâmica é baseada na fila modelo da qual se deriva, mas não é necessariamente a mesma; consulte nota 1.

O ID do usuário que o Gerenciador de Filas utiliza para as verificações de autoridade é obtido do sistema operacional. O ID do usuário é obtido quando o aplicativo estabelece conexão com o gerenciador de filas. Um aplicativo adequadamente autorizado pode emitir uma chamada MQOPEN especificando um ID do usuário alternativo; as verificações de controle de acesso são então feitas no ID do usuário alternativo. Usar um ID de usuário alternativo não muda o ID de usuário associado ao aplicativo, apenas o ID que é usado para verificações de controle de acesso.

Quando um aplicativo é subscrito para um tópico usando uma chamada MQSUB

Quando um aplicativo é subscrito para um tópico, ele especifica o tipo de operação que precisa executar. Ele estará criando uma assinatura, alterando uma assinatura existente ou continuando uma assinatura existente sem mudá-la. Para cada tipo de operação, o gerenciador de filas verifica se o ID do usuário que está associado ao aplicativo possui a autoridade para executar a operação.

Quando um aplicativo é subscrito para um tópico, as verificações de autoridade são executadas com relação a objetos de tópicos que estão localizados na árvore de tópicos. Os objetos do tópico estão em, ou acima do ponto na árvore de tópicos na qual o aplicativo foi inscrito. As verificações de autoridade podem envolver verificações em mais de um objeto de tópico.O ID do usuário que o Gerenciador de Filas utiliza para as verificações de autoridade é obtido do sistema operacional. O ID do usuário é obtido quando o aplicativo estabelece conexão com o gerenciador de filas.

O gerenciador de filas executa verificações de autoridade em filas de assinantes, mas não em filas gerenciadas.

Quando um aplicativo exclui uma fila dinâmica permanente usando uma chamada MQCLOSE

A manipulação de objetos especificada na chamada MQCLOSE não é necessariamente a mesma retornada pela chamada MQOPEN que criou a fila dinâmica permanente. Se for diferente, o gerenciador de filas verifica o ID do usuário associado ao aplicativo que emitiu a chamada MQCLOSE. Ele verifica se o ID de usuário tem autorização para excluir a fila.

Quando um aplicativo que fecha uma assinatura para removê-la não a criou, a autoridade apropriada é requerida para removê-la.

Quando um comando PCF que opera em um objeto do WebSphere MQ é processado pelo servidor de comandos

Essa regra inclui o caso em que um comando PCF opera em um objeto de informações sobre autenticação.

O ID do usuário utilizado para as verificações de autoridade é o encontrado no campo `UserIdentifier` , no descritor de mensagens do comando PCF. Esse ID do usuário deve ter as autoridades requeridas no gerenciador de fila em que o comando é processado. O comando MQSC equivalente encapsulado dentro de um comando PCF Escape é tratado da mesma forma. Para obter mais informações sobre o campo `UserIdentifier` e como ele está definido, consulte “Contexto da mensagem” na página 53.

Alternar autoridade do usuário

Quando um aplicativo abre um objeto ou assina um tópico, o aplicativo pode fornecer um ID de usuário na chamada MQOPEN, MQPUT1 ou MQSUB. Ele pode solicitar ao gerenciador de filas para usar esse ID de usuário para verificações de autoridade, em vez daquele associado ao aplicativo.

O aplicativo será bem-sucedido ao abrir o objeto somente se as duas condições a seguir forem atendidas:

- O ID do usuário associado ao aplicativo tem autoridade para fornecer um ID de usuário diferente para verificações de autoridade. Diz-se que o aplicativo tem *autoridade de usuário alternativo*.
- O ID do usuário fornecido pelo aplicativo possui a autoridade para abrir o objeto para os tipos de operações solicitadas ou para assinar o tópico.

Contexto da mensagem

As informações sobre *contexto da mensagem* permitem que o aplicativo que recupera uma mensagem descubra o emissor dela. As informações ficam retidas em campos no descritor de mensagens e os campos são divididos em três partes lógicas

Essas partes são as seguintes:

contexto de identidade
Estes campos contêm informações sobre o usuário do aplicativo que colocou a mensagem na fila.

contexto de origem
Estes campos contêm informações sobre o aplicativo em si e quando foi que a mensagem foi colocada na fila.

contexto do usuário
Estes campos contêm propriedades de mensagens que os aplicativos utilizam para selecionar as mensagens que o gerenciador de filas deve entregar.

Quando um aplicativo coloca uma mensagem em uma fila, pode solicitar que o gerenciador de fila gere as informações de contexto na mensagem. Esta é a ação padrão. Alternativamente, ele pode especificar que os campos de contexto não contenham informações. O ID do usuário associado a um aplicativo não requer autoridade especial para nenhum desses.

Um aplicativo pode definir os campos de contexto de identidade em uma mensagem, permitindo que o gerenciador de fila gere o contexto de origem ou pode definir todos os campos de contexto. Pode também passar os campos de contexto de identidade de uma mensagem recuperada para uma mensagem que esteja colocando na fila ou passar todos os campos de contexto. Porém, o ID do usuário associado a um aplicativo requer autoridade para definir ou passar as informações de contexto. Um aplicativo especifica se pretende definir ou passar informações de contexto quando abrir a fila na qual está para colocar mensagens e sua autoridade é verificada nesse momento.

A seguir, uma descrição breve de cada campo de contexto:

Contexto de Identidade

UserIdentifier
O ID do usuário associado ao aplicativo que colocou a mensagem. Se o gerenciador de fila definir este campo, ele será definido com o ID obtido do sistema operacional de quando o aplicativo estabelece conexão com o gerenciador de fila.

AccountingToken
As informações podem ser utilizadas para cobrar o trabalho feito como resultado da mensagem.

ApplIdentityData
Se o ID do usuário associado ao aplicativo tiver autoridade para definir os campos de contexto de identidade ou todos os campos de contexto, o aplicativo poderá definir esse campo com qualquer valor relacionado à identidade. Se o gerenciador de fila definir esse campo, será deixado em branco.

Contexto de origem

PutApplType
O tipo do aplicativo que colocou a mensagem; uma transação do CICS , por exemplo

PutApplName
O nome do aplicativo que coloca a mensagem.

PutDate
A data em que a mensagem foi colocada.

PutTime
A hora em que a mensagem foi colocada.

ApplOriginData
Se o ID do usuário associado ao aplicativo tiver autoridade para definir todos os campos de contexto, o aplicativo poderá definir esse campo com qualquer valor relacionado à origem. Se o gerenciador de fila definir esse campo, será deixado em branco.

Contexto de Usuário

Os seguintes valores são suportados para **MQINQMP** ou **MQSETMP**:

MQPD_USER _CONTEXT

A propriedade é associada com o contexto do usuário.

Não é necessária nenhuma autorização especial para poder definir uma propriedade associada ao contexto do usuário utilizando a chamada MQSETMP.

Em um gerenciador de filas V7.0 ou subsequente, uma propriedade associada ao contexto de usuário é salva conforme descrito para MQOO_SAVE_ALL_CONTEXT. Um MQPUT com MQOO_PASS_ALL_CONTEXT especificado faz com que a propriedade seja copiada do contexto salvo para a nova mensagem.

MQPD_NO_CONTEXT

A propriedade não é associada com um contexto de mensagem.

Um valor não reconhecido é rejeitado com um MQRC_PD_ERROR. O valor inicial deste campo é **MQPD_NO_CONTEXT**.

Para obter uma descrição detalhada de cada um dos campos de contexto, consulte MQMD - Descritor de mensagens. Para obter mais informações sobre como usar o contexto da mensagem, consulte Contexto da Mensagem.

Autoridade para trabalhar com objetos IBM WebSphere MQ nos sistemas UNIX, Linux e Windows

O componente de serviço de autorização fornecido com o IBM WebSphere MQ é chamado de *gerenciador de autoridade de objeto (OAM)*. Ele fornece controle de acesso por meio de verificações de autenticação e autorização.

1. Autenticação.

   A verificação de autenticação executada pelo OAM fornecido com o IBM WebSphere MQ é básico, e é executada apenas em circunstâncias específicas. Seu objetivo não é atender aos requisitos rígidos esperados em um ambiente altamente seguro.

   O OAM executa sua verificação de autenticação quando um aplicativo se conecta a um gerenciador de filas e as condições a seguir são verdadeiras.

   Se uma estrutura MQCSP tiver sido fornecida pelo aplicativo de conexão e o atributo *AuthenticationType* na estrutura MQCSP receber o valor MQCSP_AUTH_USER_ID_AND_PWD, a verificação será executada pelo OAM em sua função MQZID_AUTHENTICATE_USER. Essa é a verificação: o ID do usuário na estrutura MQCSP é comparado com o ID do usuário no *IdentityContext* (MQZIC), para determinar se eles correspondem. Se não corresponderem, a verificação falhará.

   Essa verificação básica não pretende se uma autenticação completa do usuário. Por exemplo, não há nenhuma verificação da autenticidade do usuário ao verificar a senha fornecida na estrutura MQCSP. Além disso, se o aplicativo omitir uma estrutura MQCSP, nenhuma verificação será executada.

   Se os serviços de autenticação mais completos são requeridos no gerenciador de filas por meio do componente de serviço de autorização, o OAM fornecido com o IBM WebSphere MQ não oferece isto. Você deverá gravar um novo componente de serviço de autorização ou obter um junto a um fornecedor.

2. Autorização.

   As verificações de autorização são abrangentes e seu objetivo é atender a requisitos mais normais.

   As verificações de autorização são executadas quando um aplicativo emite uma chamada MQI para acessar um gerenciador de filas, uma fila, um processo, um tópico ou uma lista de nomes. Elas também são executadas em outros momentos, por exemplo, quando um comando está sendo executado pelo Servidor de Comandos.

Nos sistemas UNIX, Linux e Windows, o *serviço de autorização* fornece o controle de acesso quando um aplicativo emite uma chamada de MQI para acessar um objeto do IBM WebSphere MQ que é um gerenciador de filas, fila, processo, tópico, ou lista de nomes. Isso inclui verificações da autoridade do usuário alternativo e a autoridade para definir ou passar informações de contexto.

No Windows, o OAM fornece aos membros do grupo de Administradores a autoridade para acessar todos os objetos do IBM WebSphere MQ, mesmo quando o UAC está ativado.

Além disso, em sistemas Windows , a conta SYSTEM tem acesso total aos recursos do IBM WebSphere MQ

O serviço de autorização também fornece verificações de autoridade quando um comando de PCF opera em um desses objetos do IBM WebSphere MQ ou em um objeto de informações sobre autenticação. O comando MQSC equivalente encapsulado dentro de um comando PCF Escape é tratado da mesma forma.

O serviço de autorização é um *serviço instalável* , que significa que é implementado por um ou mais *componentes de serviços instaláveis*. Cada componente é chamado por uma interface documentada. Isto permite que usuários e fornecedores forneçam componentes para aumentar ou substituir os fornecidos pelos produtos IBM WebSphere MQ.

O componente de serviço de autorização fornecido com o IBM WebSphere MQ é chamado de *gerenciador de autoridade de objeto (OAM)*. O OAM é ativado automaticamente para cada gerenciador de fila criado.

O OAM mantém uma lista de controle de acesso (ACL) para cada objeto do IBM WebSphere MQ para o qual ele está controlando o acesso. Em sistemas UNIX and Linux, somente IDs de grupos podem aparecer em uma ACL. Isto significa que todos os membros de um grupo possuem as mesmas autoridades. Nos sistemas Windows, os IDs do usuário e os IDs de grupo podem aparecer em uma ACL. Isto significa as autoridades podem ser concedidas para usuários individuais e grupos.

Uma limitação de 12 caracteres aplica-se ao grupo e ao ID do usuário. As plataformas UNIX geralmente restringem o comprimento de um ID do usuário a 12 caracteres.AIX e Linux aumentaram esse limite, mas o IBM WebSphere MQ continua a observar uma restrição de 12 caracteres em todas as plataformas UNIX . Se você usar um ID de usuário com mais de 12 caracteres, o IBM WebSphere MQ o substitui com o valor de "UNKNOWN". Não defina um ID do usuário com um valor de "UNKNOWN".

O OAM pode autenticar um usuário e alterar os campos de contexto de identidade adequados. Você ativa isto especificando uma estrutura MQCSP (connection security parameters) em uma chamada MQCONNX. A estrutura é passada para a função de Autenticar Usuário OAM (MQZ_AUTHENTICATE_USER), que define campos de contexto de identidade adequados. Se uma conexão do MQCONNX a partir de um cliente IBM WebSphere MQ, as informações no MQCSP são fluídas para o gerenciador de filas ao qual o cliente está se conectando através do canal de conexão do cliente e de conexão do servidor. Se as saídas de segurança estiverem definidas neste canal, a MQCSP é passada para cada saída de segurança e pode ser alterada pela saída. As saídas de segurança também podem criar a MQCSP. Para obter mais detalhes do uso das saídas de segurança neste contexto, consulte Programas de saída de segurança do canal.

Nos sistemas UNIX, Linux e Windows, o comando de controle **setmqaut** concede ou revoga autoridades e é usado para manter as ACLs. Por exemplo, o comando:

```
setmqaut -m JUPITER -t queue -n MOON.EUROPA -g VOYAGER +browse +get
```

permite que membros do grupo VOYAGER procurem mensagens na fila MOON.EUROPA que é de propriedade do gerenciador de fila JUPITER. Permite também que os membros obtenham mensagens da fila. Para revogar estas autoridades posteriormente, insira o seguinte comando:

```
setmqaut -m JUPITER -t queue -n MOON.EUROPA -g VOYAGER -browse -get
```

O comando:

```
setmqaut -m JUPITER -t queue -n MOON.* -g VOYAGER +put
```

Permite que os membros do grupo VOYAGER coloquem mensagens em qualquer fila com um nome que comece com os caracteres `MOON.` MOON.* é o nome de um perfil genérico..Um *perfil genérico* permite conceder autoridades para um conjunto de objetos usando um único comando **setmqaut** .

O comando de controle **dspmqaut** está disponível para exibir as autoridades atuais que um usuário ou um grupo tem para um objeto especificado.O comando de controle **dmpmqaut** também está disponível para exibir as autoridades atuais associadas aos perfis genéricos.

Se você não desejar nenhuma verificação de autoridade, por exemplo, em um ambiente de teste, poderá desativar o OAM.

Usando PCF para acessar comandos OAM
Nos sistemas UNIX, Linux e Windows , é possível usar comandos PCF para acessar comandos de administração do OAM.

Os comando de PCF e seus comandos OAM equivalentes são os seguintes:

Tabela 6. Comandos PCF e seus comandos OAM equivalentes	
comando PCF	**Comando OAM**
Consultar Registros de Autoridade	dmpmqaut
Solicitar Autoridade de Entidade	dspmqaut
Configurar Registro de Autoridade	setmqaut
Excluir Registro de Autoridade	setmqaut com opção -remove

Os comandos **setmqaut** e **dmpmqaut** são restritos a membros do grupo mqm. Os comandos de PCF equivalentes podem ser executados por usuários em qualquer grupo que tenha recebido as autoridades dsp e chg no gerenciador de filas.

Para obter mais informações sobre como usar esses comandos, consulte Introdução aos formatos de comando programáveis .

Segurança para o Sistema de Mensagens Remoto

Esta seção trata dos aspectos do sistema de mensagens remoto de segurança.

Você deve fornecer aos usuários a autoridade para usar os recursos do IBM WebSphere MQ. Isto está organizado de acordo com as ações a serem obtidas em relação aos objetos e definições. Por exemplo:

- Os gerenciadores de fila podem ser iniciados ou parados por usuários autorizados
- Os aplicativos devem se conectar ao gerenciador de filas e ter autoridade para usar as filas
- Os canais de mensagens devem ser criados e controlados pelos usuários autorizados
- Os objetos são mantidos nas bibliotecas e o acesso a essas bibliotecas pode estar restrito

O agente do canal de mensagem em um site remoto deve verificar se a mensagem está sendo entregue originada de um usuário com autoridade para tal neste site remoto. Além disso, como os MCAs podem ser iniciados remotamente, pode ser necessário verificar se os processos remotos que tentam iniciar os seus MCAs estão autorizados a fazê-lo. Existem quatro maneira possíveis para tratar disso:

1. Fazer uso apropriado do atributo PutAuthority da sua definição de canal RCVR, RQSTR ou CLUSRCVR para controlar qual usuário é usado para as verificações de autorização no momento em que as mensagens recebidas são colocadas nas suas filas. Consulte a descrição do comando DEFINE CHANNEL na Referência de Comandos MQSC.
2. Implementar os registros de autenticação de canal para rejeitar as tentativas de conexão indesejadas ou para configurar um valor MCAUSER com base no seguinte: endereço IP remoto, ID do usuário remoto, SSL ou Nome Distinto (DN) do Assunto TLS ou nome do gerenciador de filas remotas.
3. Implemente a verificação de segurança de *saída de usuário* verificando se o canal de mensagem correspondente está autorizado. A segurança da instalação que hospeda o canal correspondente assegura que todos os usuários estejam corretamente autorizados, de modo que você não precise verificar as mensagens individuais.
4. Implemente o processamento de mensagens de *saída de usuário* para assegurar que as mensagens individuais sejam examinadas para autorização.

Segurança de objetos em sistemas UNIX and Linux

Os usuários de Administração devem ser parte do grupo mqm em seu sistema (incluindo raiz), se este ID usar os comandos de administração do IBM WebSphere MQ.

É necessário sempre executar amqcrsta como o ID do usuário "mqm".

IDs do usuário em sistemas UNIX and Linux ..

O gerenciador de filas converte todos os identificadores com letras maiúsculas ou com letras maiúsculas e minúsculas em identificadores com letras minúsculas. O gerenciador de filas, então, insere os identificadores de usuário no contexto da parte de uma mensagem, ou verifica sua autorização. As autorizações são, portanto, baseadas apenas em identificadores com letras minúsculas.

Segurança de Objetos em Sistemas Windows

Os usuários de administração devem fazer parte do grupo mqm e do grupo de administradores nos sistemas Windows se esse ID for usar comandos de administração IBM WebSphere MQ .

IDs de usuário em sistemas Windows

Nos sistemas Windows , *se não houver nenhuma saída de mensagem instalada*, o gerenciador de filas converte quaisquer identificadores de usuário maiúsculos ou compostos por letras maiúsculas e minúsculas em minúsculas. O gerenciador de filas, então, insere os identificadores de usuário no contexto da parte de uma mensagem, ou verifica sua autorização. As autorizações são, portanto, baseadas apenas em identificadores com letras minúsculas.

IDs do usuário em sistemas

Plataformas diferentes de Windows , UNIX and Linux sistemas usam caracteres maiúsculos para IDs do usuário em mensagens..

Para permitir que os sistemas Windows, UNIX and Linux usem IDs de usuário minúsculos em mensagens, as conversões a seguir são realizadas pelo agente do canal de mensagem (MCA) nestas plataformas:

Na extremidade de envio
Os caracteres alfabéticos em todos os IDs do usuário são convertidos em caracteres maiúsculos, se não houver nenhuma saída de mensagem instalada.

Na extremidade de recebimento
Os caracteres alfabéticos em todos os IDs de usuário são convertidos em caracteres minúsculos, se nenhuma saída de mensagem instalada.

As conversões automáticas não serão executadas se você fornecer uma saída de mensagem nos sistemas UNIX, Linux e Windows por qualquer outro motivo

Usando um serviço de autorização customizado

O IBM WebSphere MQ fornece um serviço de autorização instalável. É possível escolher instalar um serviço alternativo.

O componente de serviço de autorização fornecido com o IBM WebSphere MQ é chamado Gerenciador de Autoridade de Objeto (OAM). Se o OAM não fornecer as instalações de autorização necessárias, será possível gravar seu próprio componente de serviço de autorização. As funções do serviço instalável, que deve ser implementado por um componente de serviço de autorização, são descritas em Informações de referência da interface de serviços instaláveis.

Controle de acesso para clientes

O controle de acesso é baseado nos IDs de usuário. Pode haver muitos IDs de usuário para administrar, e os IDs de usuário podem estar em diferentes formatos. É possível configurar a propriedade MCAUSER do canal de conexão do servidor com um valor de ID do usuário especial para uso pelos clientes.

O controle de acesso em IBM WebSphere MQ é baseado nos IDs do usuário. O ID do usuário do processo que faz chamadas MQI é normalmente usado. Para os clientes do MQ MQI, o MCA de conexão do servidor

faz chamadas MQI em nome de clientes do MQ MQI. É possível selecionar um ID do usuário alternativo para a conexão do servidor MCA para usar para fazer chamadas MQI. O ID do usuário alternativo pode ser associado à estação de trabalho do cliente ou a qualquer coisa escolhida para organizar e controlar o acesso dos clientes. O ID do usuário precisa ter as autoridades necessárias alocadas para ele no servidor para emitir chamadas MQI. Escolher um ID do usuário alternativo é preferível a permitir que clientes façam chamadas MQI com a autoridade da conexão do servidor MCA.

Tabela 7. O ID do usuário usado por um canal de conexão do servidor

ID do usuário	Quando usado
O ID do usuário que é configurado por uma saída de segurança	Usado a menos que seja bloqueado por uma regra **CHLAUTH TYPE(BLOCKUSER)**. Consulte a seção a seguir, "Configurando o ID do usuário em uma saída de segurança" na página 60, para obter mais informações.
O ID do usuário que é configurado por uma regra CHLAUTH	Usado a menos que substituído por uma saída de segurança. Consulte Registros de autenticação de canal para obter mais informações.
O ID do usuário que é definido no atributo **MCAUSER** na definição de canal SVRCONN	Usado a menos que substituído por uma saída de segurança ou uma regra CHLAUTH.
O ID do usuário que é transmitido a partir da máquina cliente	Usado quando nenhum ID usado é configurado por qualquer outro meio
O ID do usuário que iniciou o canal de conexão do servidor	Usado quando nenhum ID do usuário é configurado por qualquer outro meio e nenhum ID do usuário cliente é transmitido. Consulte a seção a seguir, "O ID do usuário que executa o programa do canal" na página 60 para obter mais informações.

Como o MCA de conexão do servidor faz chamadas de MQI em nome de usuários remotos, é importante considerar as implicações de segurança do MCA de conexão do servidor que emite chamadas de MQI em nome de clientes remotos, e como administrar o acesso de um número potencialmente grande de usuários.

- Uma abordagem é para a conexão do servidor MCA emitir chamadas MQI em sua própria autoridade. Mas tenha cuidado, normalmente é indesejável para a conexão do servidor MCA, com seus recursos de acesso poderosos, emitir chamadas MQI em nome de usuários clientes.
- Outra abordagem é usar o ID do usuário que flui a partir do cliente. A conexão do servidor MCA pode emitir chamadas MQI usando os recursos de acesso do ID do usuário do cliente. Esta abordagem apresenta várias questões a considerar:
  1. Existem diferentes formatos para o ID do usuário em diferentes plataformas. Isto às vezes causa problemas se o formato do ID do usuário no cliente diferir dos formatos aceitáveis no servidor.
  2. Há potencialmente muitos clientes com IDs de usuário diferentes e em mudança. Os IDs precisam ser definidos e gerenciados no servidor.
  3. O ID do usuário é confiável? Qualquer ID do usuário pode fluir a partir de um cliente, não necessariamente o ID do usuário que efetuou logon. Por exemplo, o cliente pode fluir um ID com total autoridade `mqm` que foi intencionalmente definida apenas no servidor por razões de segurança.
- A abordagem preferencial é definir tokens de identificação de cliente no servidor e, portanto, limitar os recursos de aplicativos conectados pelo cliente. Isto é geralmente feito configurando a propriedade MCAUSER do canal de conexão do servidor com um valor de ID do usuário especial a ser usado pelos clientes, e definindo alguns IDs para uso por clientes com nível diferente de autorização no servidor.

Configurando o ID do usuário em uma saída de segurança

Para clientes MQI do IBM WebSphere MQ , o processo que emite as chamadas MQI é a conexão do servidor MCA. O ID do usuário usado pela conexão do servidor MCA está contido nos campos `MCAUserIdentifier` ou `LongMCAUserIdentifier` do MQCD. O conteúdo destes campos é configurado por:

- Qualquer valor configurado pelas saídas de segurança
- O ID do usuário do cliente
- MCAUSER (na definição do canal de conexão do servidor)

A saída de segurança pode substituir os valores que estão visíveis para ela, quando ela é invocada.

- Se o atributo MCAUSER do canal de conexão do servidor estiver configurado como não-em branco, o valor MCAUSER será usado.
- Se o atributo MCAUSER do canal de conexão do servidor estiver em branco, o ID do usuário recebido do cliente será usado.
- Se o atributo MCAUSER do canal de conexão do servidor estiver em branco, e nenhum ID do usuário for recebido do cliente, o ID do usuário que iniciou o canal de conexão do servidor será usado.

Certifique-se de que o campo MCAUSER esteja restrito a 12 caracteres nas plataformas Windows, pois quaisquer caracteres extras serão truncados, o que pode levar a falhas de autorização

O cliente do IBM WebSphere MQ não flui o ID do usuário declarado para o servidor quando uma saída de segurança do lado do cliente está em uso.

O ID do usuário que executa o programa do canal

Quando os campos de ID do usuário forem derivados do ID do usuário que iniciou o canal de conexão do servidor, o seguinte valor será usado:

- Para o z/OS, o ID do usuário designado à tarefa iniciada pelo inicializador de canais pela tabela de procedimentos iniciados do z/OS.
- Para TCP/IP (nãoz/OS), o ID do usuário da entrada `inetd.conf` ou o ID do usuário que iniciou o listener.
- Para SNA (nãoz/OS), o ID do usuário da entrada do Servidor SNA ou (se não houver nenhum) o pedido de conexão recebido ou o ID do usuário que iniciou o listener.
- Para o NetBIOS ou SPX, o ID do usuário que iniciou o listener.

Se qualquer definição de canal de conexão do servidor existir tendo o atributo MCAUSER configurado como em branco, os clientes poderão usar esta definição de canal para se conectar ao gerenciador de filas com autoridade de acesso determinada pelo ID do usuário fornecido pelo cliente. Pode haver uma exposição de segurança se o sistema em que o gerenciador de filas está sendo executado permitir conexões de rede não-autorizadas. O canal de conexão do servidor padrão IBM WebSphere MQ (SYSTEM.DEF.SVRCONN) possui o atributo MCAUSER configurado para em branco. Para evitar acesso não autorizado, atualize o atributo MCAUSER da definição padrão com um ID do usuário que não possui acesso aos objetos do IBM WebSphere MQ MQ.

Caso de IDs de usuário

Quando você define um canal com `runmqsc`, o atributo MCAUSER é alterado para letra maiúscula a menos que o ID do usuário esteja contido entre aspas simples.

Para servidores nos sistemas UNIX, Linux e Windows, o conteúdo do campo `MCAUserIdentifier` que é recebido do cliente é mudado para letras minúsculas.

Para servidores no IBM i, o conteúdo do campo `LongMCAUserIdentifier` que é recebido do cliente é mudado para letras maiúsculas.

Para servidores nos sistemas UNIX and Linux, o conteúdo do campo `LongMCAUserIdentifier` que é recebido do cliente é mudado para letras minúsculas.

Por padrão, o ID do usuário que é transmitido quando um aplicativo de ligação do MQ JMS é usado é o ID de usuário para a JVM na qual o aplicativo está executando.

Também é possível transmitir um ID de usuário por meio do método `createQueueConnection`.

Planejando a confidencialidade

Planeje como manter seus dados confidenciais.

É possível implementar confidencialidade no nível do aplicativo ou no nível de link. É possível escolher usar SSL ou TLS nos casos em que se deve planejar o uso de certificados digitais. Também é possível usar programas de saída de canal se os recursos padrão não atenderem aos seus requisitos.

Conceitos relacionados

"Comparando a segurança no nível do link com a segurança no nível do aplicativo" na página 61
Este tópico contém informações sobre os diversos aspectos da segurança em nível de aplicativo e segurança em nível de link, e compara os dois níveis de segurança.

"Programas de Saída de Canal" na página 66
Os *programas de saída de Canal* são programas chamados em lugares definidos na seqüência do processamento de um MCA. Usuários e fornecedores podem desenvolver seus próprios programas de saída de canal. Alguns são fornecidos pela IBM

"Protegendo canais com SSL" na página 73
O suporte SSL no IBM WebSphere MQ usa o objeto de informações de autenticação do gerenciador de filas e vários comandos do MQSC Também se deve considerar o uso de certificados digitais.

Comparando a segurança no nível do link com a segurança no nível do aplicativo

Este tópico contém informações sobre os diversos aspectos da segurança em nível de aplicativo e segurança em nível de link, e compara os dois níveis de segurança.

A segurança em nível de link e de aplicativo é ilustrada na Figura 8 na página 61.

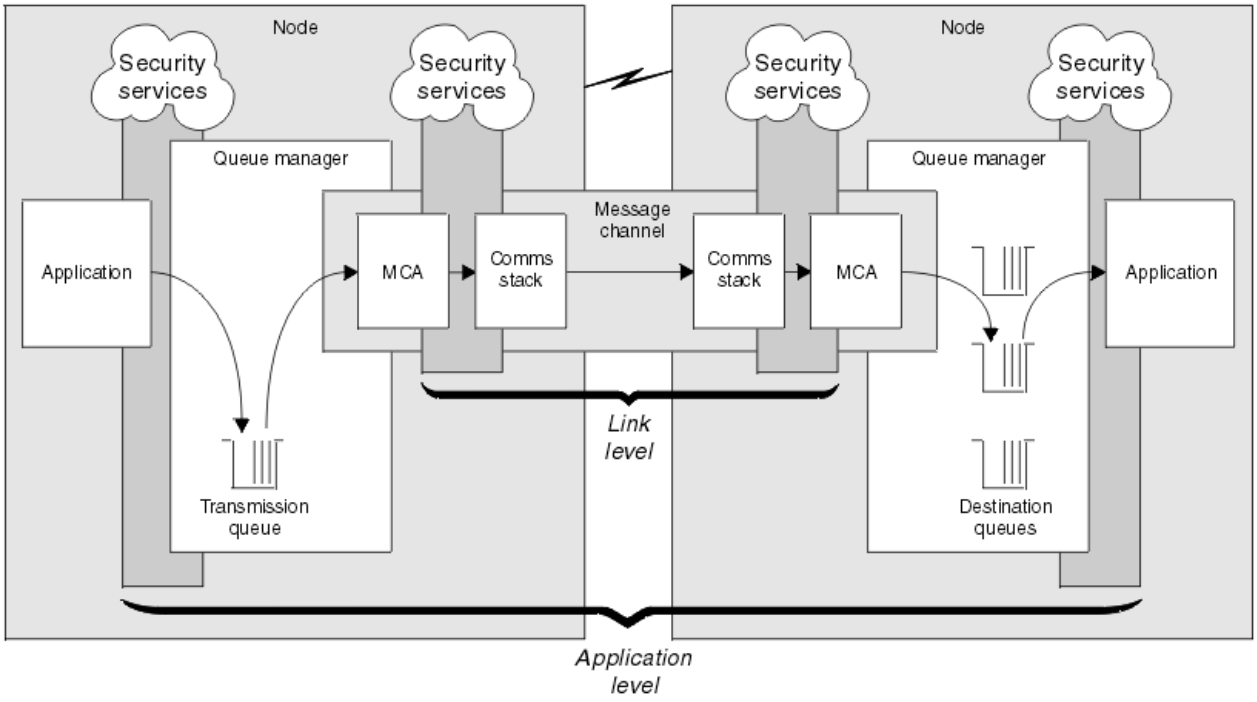


Figura 8. Segurança no nível do link e segurança no nível do aplicativo

Protegendo mensagens em filas

A segurança no nível do link pode proteger mensagens enquanto são transferidas de um gerenciador de filas para outro. É de particular importância quando as mensagens são transmitidas através de uma rede não protegida. Ela não pode, porém, proteger as mensagens enquanto elas estão armazenadas em filas em um gerenciador de filas de origem, de destino ou intermediário.

A segurança em nível de aplicativo, em comparação, pode proteger as mensagens enquanto elas estão armazenadas em filas e se aplica mesmo quando não está sendo utilizado enfileiramento distribuído. Essa é a principal diferença entre a segurança no nível do link e a segurança no nível do aplicativo, e é ilustrada na Figura 8 na página 61.

Gerenciadores de filas que não estão executando em ambientes controlados e de confiança

Se um gerenciador de filas estiver em execução em um ambiente controlado e confiável, os mecanismos de controle de acesso fornecidos pelo WebSphere MQ podem ser considerados suficientes para proteger as mensagens armazenadas em suas fila. Isso é especialmente verdadeiro se somente filas locais estiverem envolvidas e as mensagens nunca deixarem o gerenciador de filas. A segurança no nível do aplicativo neste caso pode ser considerada desnecessária.

Ela também pode ser considerada desnecessária se as mensagens forem transferidas para outro gerenciador de filas que também esteja executando em um ambiente controlado e de confiança, ou forem recebidas de um gerenciador de filas nessas condições. A necessidade de segurança em nível de aplicativo torna-se maior quando as mensagens são transferidas ou recebidas de um gerenciador de filas que não está sendo executado em um ambiente controlado e confiável.

Diferenças de custo

A segurança no nível do aplicativo pode custar mais que a segurança no nível do link em termos de administração e de desempenho.

É provável que o custo da administração seja maior porque existem mais restrições para configurar e manter. Por exemplo, você pode precisar assegurar que um determinado usuário envia somente certos tipos de mensagens e envia mensagens somente para certos destinos. Por outro lado, você pode precisar assegurar que um determinado usuário recebe somente certos tipos de mensagens e recebe mensagens somente de certas origens. Em vez de gerenciar os serviços de segurança no nível do link em um único canal de mensagem, você pode precisar configurar e manter regras para cada par de usuários que trocam mensagens através desse canal.

Pode haver um impacto no desempenho se os serviços de segurança forem chamados sempre que um aplicativo colocar ou obtiver uma mensagem.

As organizações tendem a considerar a segurança no nível do link primeiro porque ela pode ser mais fácil de implementar. Elas consideram a segurança em nível de aplicativo se descobrem que a segurança no nível do link não satisfaz todos os seus requisitos.

Disponibilidade de componentes

Geralmente, em um ambiente distribuído, um serviço de segurança requer um componente em pelo menos dois sistemas. Por exemplo, uma mensagem pode ser criptografada em um sistema e descriptografada em outro. Isso se aplica tanto à segurança no nível do link quanto à segurança em nível de aplicativo.

Em um ambiente heterogêneo, com diferentes plataformas em uso, cada uma delas com deferentes níveis de funções de segurança, os componentes necessários de um serviço de segurança podem não estar disponíveis para todas as plataformas nas quais eles são necessários e de uma forma fácil de utilizar. Isso provavelmente é um problema maior para a segurança em nível de aplicativo que para a segurança no nível do link, especialmente se você pretender fornecer sua própria segurança em nível de aplicativo comprando componentes de várias origens.

Mensagens em uma fila de cartas não entregues

Se uma mensagem for protegida pela segurança em nível de aplicativo, pode haver um problema se, por algum motivo, a mensagem não atingir seu destino e for colocada em uma fila de cartas não entregues. Se você não descobrir como processar a mensagem a partir das informações no descritor da mensagem e do cabeçalho da carta não entregue, poderá precisar inspecionar o conteúdo dos dados do aplicativo. Não é possível fazer isso se os dados do aplicativo estiverem criptografados e somente o destinatário pretendido poderá decriptografá-los.

O que a segurança em nível de aplicativo não pode fazer

A segurança no nível do aplicativo não é uma solução completa. Mesmo se você implementar a segurança em nível de aplicativo, alguns serviços da segurança no nível do link ainda podem ser necessários. Por exemplo:

- Quando um canal inicia, a autenticação mútua dos dois MCAs pode ainda ser uma exigência. Isso pode ser feito somente por um serviço de segurança no nível do link.
- A segurança em nível de aplicativo não pode proteger o cabeçalho da fila de transmissão, MQXQH, o qual inclui o descritor da mensagem incorporado. Também não é possível proteger os dados nos fluxos do protocolo de canal do WebSphere MQ que não sejam dados de mensagens.. Somente a segurança no nível do link pode fornecer essa proteção.
- Se os serviços da segurança em nível de aplicativo forem chamados na extremidade do servidor de um canal MQI, os serviços não poderão proteger os parâmetros das chamadas de MQI que forem enviadas pelo canal. Em particular, os dados do aplicativo em um chamada MQPUT, MQPUT1 ou MQGET não são protegidos. Somente a segurança no nível do link pode fornecer a proteção neste caso.

Segurança no nível do link

Segurança no nível do link se refere ao serviços de segurança que são chamados, direta ou indiretamente, por um MCA, pelo subsistema de comunicações ou por uma combinação dos dois trabalhando em conjunto.

A segurança em nível de link é ilustrada no Figura 8 na página 61.

Eis alguns exemplos de serviços de segurança no nível do link:

- O MCA em cada extremidade de um canal de mensagem pode autenticar seu parceiro. Isso é feito quando o canal iniciar e uma conexão de comunicações tiver sido estabelecido, mas antes que as mensagens comecem a fluir. Se a autenticação falhar em qualquer das extremidades, o canal será fechado e nenhuma mensagem será transferida. Este é um exemplo de um serviço de identificação e autenticação.
- Uma mensagem pode ser criptografada na extremidade de envio de um canal e decriptografada na extremidade de recepção. Este é um exemplo de um serviço de confidencialidade.
- Uma mensagem pode ser verificada na extremidade de recepção de um canal para determinar se seu conteúdo foi modificado deliberadamente enquanto ela estava sendo transmitida pela rede. Este é um exemplo de um serviço de integridade de dados.

A segurança em nível de link fornecida pelo IBM WebSphere MQ

O principal meio de provisão de confidencialidade e integridade de dados no IBM WebSphere MQ é pelo uso de SSL ou TLS. Para obter mais informações sobre o uso de SSL e TLS no IBM WebSphere MQ, consulte “IBM WebSphere MQ Suporte para SSL e TLS” na página 23. Para autenticação, o IBM WebSphere MQ fornece o recurso para usar registros de autenticação de canal. Os registros de autenticação de canal oferecem controle preciso sobre o acesso concedido à conexão de sistemas, no nível de canais individuais ou grupos de canais. Para obter mais informações, consulte “Registros de Autenticação de Canal” na página 40.

Fornecendo sua Própria Segurança em Nível de Link
Esta coleção de tópicos descreve como é possível fornecer seus próprios serviços de segurança em nível de link. Gravar seus próprios programas de saída do canal é a principal forma de fornecer seus próprios serviços de segurança em nível de link.

Os programas de saída de canal são apresentados em "Programas de Saída de Canal" na página 66. O mesmo tópico também descreve o programa de saída do canal fornecido com IBM WebSphere MQ for Windows (o programa de saída do canal SSPI). Este programa de saída de canal é fornecido em formato de fonte para que você possa alterar o código-fonte para se adequar a suas necessidades. Se este programa de saída do canal, ou programas de saída do canal disponíveis a partir de outros fornecedores, não atenderem seus requisitos, será possível projetar e gravar seu próprio. Este tópico sugere maneiras nas quais os programas de saída do canal podem fornecer serviços de segurança. Para obter informações sobre como gravar um programa de saída do canal, consulte o Gravando programas de saída do canal.

Segurança em Nível de Link Usando uma Saída de Segurança
As saídas de segurança normalmente trabalham em pares; uma em cada extremidade de um canal. Elas são chamadas imediatamente após a conclusão da negociação de dados inicial na inicialização do canal.

Saídas de segurança podem ser usadas para fornecer identificação e autenticação, controle de acesso e confidencialidade.

Segurança em Nível de Link Usando uma Saída de Mensagem
Uma saída de mensagem pode ser utilizada apenas em um canal de mensagem, não em um canal MQI. Ela tem acesso ao cabeçalho da fila de transmissão, MQXQH, que inclui o descritor de mensagens internas e os dados do aplicativo em uma mensagem. Pode modificar o conteúdo da mensagem e alterar seu comprimento.

Uma saída de mensagem pode ser utilizada para qualquer objetivo que exija acesso à mensagem inteira, em vez de uma parte dela.

Saídas de mensagem podem ser usadas para fornecer identificação e autenticação, controle de acesso, confidencialidade, integridade de dados e irrecusabilidade, e por outros motivos que não segurança.

Segurança em Nível de Link Usando Saídas de Envio e Recebimento
As saídas de envio e recebimento podem ser utilizadas nos canais de mensagem e MQI. Podem ser chamadas para todos os tipos de dados que passam em um canal e para fluxos nas duas direções.

As saídas de envio e recebimento têm acesso a cada segmento de transmissão. Elas podem modificar seu conteúdo e alterar seu comprimento.

Em um canal de mensagens, se um MCA tem que dividir uma mensagem e enviá-la em mais de um segmento de transmissão, uma saída de envio poderá ser chamada para cada segmento de transmissão que contém uma parte da mensagem e, na extremidade de recebimento, uma saída de recebimento é chamada para cada segmento de transmissão. O mesmo ocorrerá em um canal MQI se os parâmetros de entrada ou saída de uma chamada MQI forem muito grandes para serem enviados em um único segmento de transmissão.

Em um canal MQI, o byte 10 de um segmento de transmissão identifica a chamada MQI e indica se o segmento de transmissão contém os parâmetros de entrada ou saída da chamada. As saídas de envio e recebimento podem examinar este byte para determinar se a chamada MQI contém dados do aplicativo que podem necessitar de proteção.

Quando uma saída de usuário é chamada pela primeira vez, para adquirir e inicializar os recursos de que necessita, pode solicitar que o MCA reserve uma quantidade específica de espaço no buffer que contém um segmento de transmissão. Quando é chamada posteriormente para processar um segmento de transmissão, ela pode usar esse espaço para incluir uma chave criptografada ou uma assinatura digital, por exemplo. A saída de recepção correspondente na outra extremidade do canal pode remover os dados incluídos pela saída de envio e utilizá-los para processar o segmento de transmissão.

As saídas de envio e recebimento são mais adequadas para propósitos em que não precisam entender a estrutura dos dados que estão manipulando, podendo, assim, tratar cada segmento de transmissão como um objeto binário.

As saídas de envio e recebimento podem ser usadas para fornecer confidencialidade e integridade de dados, e para outros usos que não segurança.

Tarefas relacionadas

Identificando uma chamada de API em um programa de saída de envio ou recebimento

Segurança em Nível de Aplicativo

Segurança no nível do aplicativo se refere aos serviços de segurança que são chamados na interface entre um aplicativo e um gerenciador de filas ao qual ele está conectado.

Esses serviços são chamados quando o aplicativo emite chamadas de MQI para o gerenciador de filas. Os serviços podem ser chamados, direta ou indiretamente, pelo aplicativo, pelo gerenciador de filas, por outro produto que suporte o WebSphere MQ, ou por uma combinação de qualquer um deles trabalhando em conjunto A segurança em nível de aplicativo é ilustrada na Figura 8 na página 61.

A segurança em nível de aplicativo também é conhecida como *segurança ponta-a-ponta* ou *segurança no nível da mensagem*.

Eis alguns exemplos de serviços de segurança em nível de aplicativo:

- quando um aplicativo coloca uma mensagem em uma fila, o descritor da mensagem contém um ID de usuário associado ao aplicativo. Entretanto, não existem dados presentes, tais como uma senha criptografada, que possam ser utilizados para autenticar o ID do usuário. Um serviço de segurança pode incluir esses dados. Quando a mensagem for finalmente recuperada pelo aplicativo de recepção, outro componente do serviço pode autenticar o ID do usuário utilizando os dados que foram enviados com a mensagem.Este é um exemplo de um serviço de identificação e autenticação.
- Uma mensagem pode ser criptografada quando é colocada em uma fila por um aplicativo, e descriptografada quando é recuperada pelo aplicativo de recepção.Este é um exemplo de um serviço de confidencialidade.
- Uma mensagem pode ser verificada quando é recuperada pelo aplicativo de recepção. Essa verificação determina se seu conteúdo foi modificado deliberadamente desde que foi colocada pela primeira vez em uma fila pelo aplicativo de envio.Este é um exemplo de um serviço de integridade de dados.

Planejamento do Advanced Message Security

IBM WebSphere MQ Advanced Message Security (AMS) é um componente licenciado separadamente do IBM WebSphere MQ que fornece um alto nível de proteção para dados sensíveis que fluem pela rede do IBM WebSphere MQ , enquanto não impacta os aplicativos finais.

Se você estiver movendo informações altamente sigilosas ou de valor, informações especialmente confidenciais ou relacionadas a pagamento, como registros de paciente ou detalhes de cartão de crédito, deverá prestar muita atenção à segurança de informações. Assegurar que as informações que passam pela empresa retenham sua integridade e sejam protegidas contra acesso não autorizado é um desafio e uma responsabilidade contínua. É provável também que você seja obrigado a cumprir os regulamentos de segurança, sob o risco de penalidades por falta de conformidade.

É possível desenvolver suas próprias extensões de segurança para o IBM WebSphere MQ. No entanto, essas soluções requerem qualificações de especialistas e podem ser complicadas e caras de se manter. O IBM WebSphere MQ Advanced Message Security ajuda a lidar com esses desafios ao mover informações acerca da empresa entre cada tipo de sistema de TI comercial virtualmente.

O IBM WebSphere MQ Advanced Message Security estende os recursos de segurança do IBM WebSphere MQ das seguintes maneiras:

- Fornece proteção de dados de ponta a ponta no nível do aplicativo para sua infraestrutura do sistema de mensagens ponto a ponto, usando criptografia ou assinatura digital de mensagens.
- Fornece segurança abrangente sem gravar código de segurança complexo ou modificar ou recompilar aplicativos existentes.
- Usa a tecnologia de Infraestrutura de Chave Pública (PKI) para fornecer autenticação, autorização, confidencialidade e serviços de integridade de dados para mensagens.
- Fornece administração de políticas de segurança para servidores mainframe e distribuídos.

- Ele suporta servidores e clientes do IBM WebSphere MQ.
- Ele se integra com o IBM WebSphere MQ Managed File Transfer para fornecer uma solução do sistema de mensagens segura de ponta a ponta.

Para obter mais informações, consulte “IBM WebSphere MQ Advanced Message Security” na página 274.

Fornecendo sua própria segurança de nível de aplicativo
Esta coleção de tópicos descreve como é possível fornecer seus próprios serviços de segurança em nível de aplicativo.

Para ajudar a implementar a segurança no nível do aplicativo, o IBM WebSphere MQ fornece duas saídas, a saída da API e a saída cruzada da API.

Essas saídas podem fornecer serviços de identificação e autenticação, de controle de acesso, de confidencialidade, de integridade de dados e de irrecusabilidade, além de outras funções não relacionadas à segurança.

Se a saída API ou a saída de cruzamento de API não for suportada em seu ambiente de sistema, você pode desejar considerar outras maneiras de fornecer sua própria segurança de nível de aplicativo. Uma maneira é desenvolver uma API de nível mais alto que encapsule a MQI. Os programadores usam, então, essa API, em vez do MQI, para gravar aplicativos do IBM WebSphere MQ.

As razões mais comuns para utilizar uma API de nível mais alto são:

- Para ocultar os recursos mais avançados da MQI dos programadores.
- Para reforçar padrões na utilização da MQI.
- Para incluir uma função à MQI. Esta função adicional pode ser serviços de segurança.

Alguns produtos de fornecedores usam esta técnica para fornecer segurança de nível do aplicativo para IBM WebSphere MQ.

Se você estiver planejando fornecer serviços de segurança desta maneira, observe o seguinte em relação à conversão de dados:

- Se um token de segurança, tal como uma assinatura digital, tiver sido incluído aos dados de aplicativo na mensagem, qualquer código executando conversão de dados deve estar ciente da presença deste token.
- Um token de segurança pode ser derivado de uma imagem binária dos dados do aplicativo. Portanto, qualquer verificação do token deve ser feita antes de converter os dados.
- Se os dados do aplicativo na mensagem tiverem sido criptografados, eles devem ser decriptografados antes da conversão de dados.

Programas de Saída de Canal

Os *programas de saída de Canal* são programas chamados em lugares definidos na seqüência do processamento de um MCA. Usuários e fornecedores podem desenvolver seus próprios programas de saída de canal. Alguns são fornecidos pela IBM

Existem diversos tipos de programas de saída de canal, mas apenas quatro têm uma função de fornecer segurança em nível de link:

- Saída de Segurança
- Saída de mensagen
- Saída de envio
- Saída de recepção

Esses quatro tipos de programas de saída de canal são ilustrados na Figura 9 na página 67 e descritos nos tópicos a seguir.

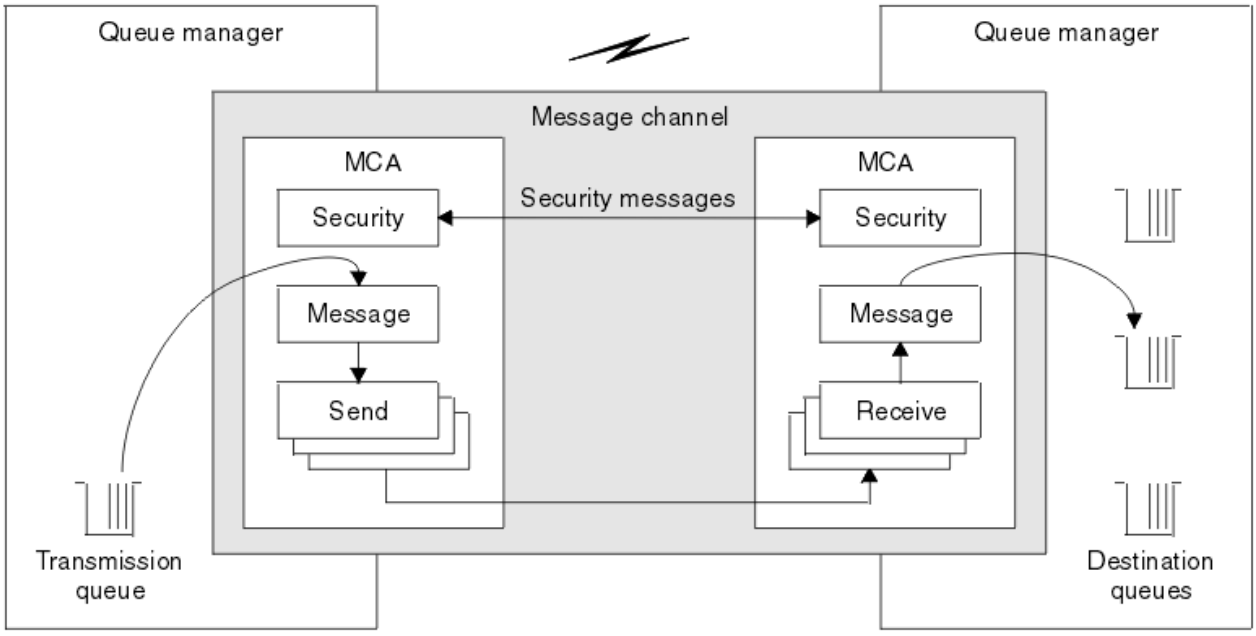


Figura 9. Saídas de segurança, mensagem, envio e recebimento em um canal de mensagens

Conceitos relacionados

Programas de Saída de Canal para Canais de Mensagens

Visão Geral da Saída de Segurança

As saídas de segurança normalmente trabalham em pares. Elas são chamadas antes dos fluxos de mensagens e seus propósitos são permitir que um MCA autentique seu parceiro.

As *Saídas de segurança* normalmente trabalham em pares; uma em cada extremidade de um canal. Elas são chamadas imediatamente após a conclusão da negociação de dados inicial na inicialização do canal, mas antes do início do fluxo de qualquer mensagem. A principal finalidade da saída de segurança é permitir ao MCA de cada extremidade de um canal autenticar o seu parceiro. No entanto, não há nada que possa evitar que uma saída de segurança efetue outra função, mesmo uma função que não tenha nada a ver com segurança.

As saídas de segurança podem se comunicar umas com as outras enviando *mensagens de segurança*. O formato de uma mensagem de segurança não é definido e é determinado pelo usuário. Um possível resultado da troca de mensagens de segurança é que uma das saídas de segurança pode decidir não prosseguir mais. Nesse caso, o canal é fechado e as mensagens não fluem. Se existir uma saída de segurança em apenas uma extremidade de um canal, a saída ainda é chamada e pode escolher se vai prosseguir ou fechar o canal.

As saídas de segurança podem ser chamadas em canais de mensagens e do MQI. O nome de uma saída de segurança é especificado como um parâmetro na definição do canal em cada extremidade de um canal.

Para obter mais informações sobre saídas de segurança, consulte “Segurança em Nível de Link Usando uma Saída de Segurança” na página 64.

Saída de mensagen

As saídas de mensagens operam apenas em canais de mensagens e normalmente funcionam em pares. Uma saída de mensagem pode operar em toda a mensagem e fazer várias mudanças nela.

As *Saídas de mensagens* nas extremidades de envio e de recebimento de um canal normalmente trabalham em pares. Uma saída de mensagens na extremidade de envio de um canal é chamada após o MCA ter recebido uma mensagem da fila de transmissão. Na extremidade de recebimento de um canal, uma saída de mensagens é chamada antes que o MCA coloque uma mensagem em sua fila de destino.

Uma saída de mensagens tem acesso ao cabeçalho da fila de transmissão, MQXQH, que inclui o descritor de mensagens embutidas, e os dados do aplicativo em uma mensagem. Uma saída de mensagens pode modificar o conteúdo da mensagem e alterar seu comprimento. Uma alteração no comprimento pode ser o resultado da compressão, descompressão, criptografia e decriptografia da mensagem. Também pode ser o resultado de incluir dados na mensagem ou remover dados dela.

As saídas de mensagens podem ser utilizadas para qualquer finalidade que exija acesso à mensagem inteira, em vez de parte dela, e não necessariamente para segurança.

Uma saída de mensagem pode determinar que a mensagem que está processando atualmente não deve continuar além da direção de seu destino. O MCA coloca a mensagem na fila dead-letter. Uma saída de mensagem pode também fechar o canal.

As saídas de mensagens podem ser chamadas apenas em canais de mensagens, não em canais do MQI. Isso ocorre porque o propósito de um canal MQI é permitir que os parâmetros de entrada e saída de chamadas MQI fluam entre o aplicativo cliente MQI do IBM WebSphere MQ e o gerenciador de filas.

O nome de uma saída de mensagens é especificado como um parâmetro na definição do canal em cada extremidade de um canal. Você também pode especificar uma lista de saídas de mensagens para serem executadas sucessivamente.

Para obter mais informações sobre saídas de mensagem, consulte “Segurança em Nível de Link Usando uma Saída de Mensagem” na página 64.

Saídas de Envio e Recebimento

As saídas de envio e recebimento geralmente trabalham em pares. Elas operam em segmentos de transmissão e são melhor usadas onde a estrutura dos dados que estão processando não for relevante.

Uma *saída de envio* em uma extremidade de um canal e uma *saída de recebimento* na outra extremidade normalmente trabalham em pares. Uma saída de envio é chamada pouco antes de um MCA emitir um envio de comunicação para enviar dados para uma conexão de comunicação. Uma saída de recebimento é chamada logo depois que um MCA recuperou o controle após um recebimento de comunicação e recebeu dados de uma conexão de comunicação. Se as conversações compartilhadas estiverem em uso sobre um canal MQI, uma instância diferente de uma saída de envio e de recebimento será chamada para cada conversação.

Os fluxos de protocolo de canais do IBM WebSphere MQ entre dois MCAs em um canal de mensagens contêm informações de controle, assim como dados das mensagens. Da mesma forma, em um canal do MQI, os fluxos contêm informações sobre controle assim como os parâmetros das chamadas do MQI. As saídas de envio e recebimento são chamadas para todos os tipos de dados.

Os dados de mensagens fluem em apenas uma direção em um canal de mensagens mas, em um canal do MQI, os parâmetros de entrada de uma chamada do MQI fluem em uma direção e os parâmetros de saída fluem na outra direção. Em canais de mensagens e do MQI, as informações de controle fluem em ambas as direções. Como resultado, as saídas de envio e recebimento podem ser chamadas em ambas as extremidades de um canal.

A unidade de dados que é transmitida em um único fluxo entre dois MCAs é denominada um *segmento de transmissão*. As saídas de envio e recebimento têm acesso a cada segmento de transmissão. Elas podem modificar seu conteúdo e alterar seu comprimento. No entanto, uma saída de envio não deve alterar os oito primeiros bytes de um segmento de transmissão. Esses 8 bytes fazem parte do cabeçalho do protocolo do canal do IBM WebSphere MQ. Também existem restrições em relação a quanto uma saída de envio pode aumentar o comprimento de um segmento de transmissão. Especificamente, uma saída de envio não pode aumentar seu comprimento além do máximo negociado entre os dois MCAs na inicialização do canal.

Em um canal de mensagens, se uma mensagem for muito grande para ser enviada em um único segmento de transmissão, o MCA de envio divide a mensagem e a envia em mais de um segmento de transmissão. Como conseqüência, uma saída de envio é chamada para cada segmento de transmissão contendo uma parte da mensagem e, na extremidade de recebimento, uma saída de recebimento é chamada para cada segmento de transmissão. O MCA de recebimento reconstitui a mensagem a partir dos segmentos de transmissão após serem processados pela saída de recebimento.

Da mesma forma, em um canal do MQI, os parâmetros de entrada ou de saída de uma chamada do MQI são enviados em mais de um segmento de transmissão se forem muito grandes. Isso pode ocorrer, por exemplo, em uma chamada MQPUT, MQPUT1 ou MQGET se os dados do aplicativo forem grandes o suficiente.

Levando em conta essas considerações, é mais apropriado utilizar saídas de envio e recebimento para objetivos nos quais elas não precisem entender a estrutura dos dados que estão tratando e possam, assim, tratar cada segmento de transmissão como um objeto binário.

Uma saída de envio ou de recebimento pode fechar um canal.

Os nomes de uma saída de envio e de uma saída de recebimento são especificados como parâmetros na definição do canal em cada extremidade de um canal. Você também pode especificar uma lista de saídas de envio a serem executadas sucessivamente. Da mesma maneira, você pode especificar uma lista de saídas de recebimento.

Para obter mais informações sobre saídas de envio e recebimento, consulte “Segurança em Nível de Link Usando Saídas de Envio e Recebimento” na página 64.

Planejando a integridade de dados

Planeje como preservar a integridade dos dados.

É possível implementar a integridade dos dados no nível do aplicativo ou no nível de link.

No nível do aplicativo, você pode optar por usar IBM WebSphere MQ Advanced Message Security para assinar digitalmente mensagens para proteger contra modificação não autorizada. Também é possível usar programas de saída de API se os recursos padrão não satisfizerem seus requisitos..

No nível de link, será possível optar por usar SSL ou TLS. Em tais casos, deve-se planejar o uso de certificados digitais. Também é possível usar programas de saída de canal se os recursos padrão não atenderem aos seus requisitos.

Conceitos relacionados

“Protegendo canais com SSL” na página 73
O suporte SSL no IBM WebSphere MQ usa o objeto de informações de autenticação do gerenciador de filas e vários comandos do MQSC Também se deve considerar o uso de certificados digitais.

“Integridade de dados no IBM WebSphere MQ ..” na página 22
É possível usar um serviço de integridade de dados para detectar se uma mensagem foi modificada.

“Planejamento do Advanced Message Security” na página 65
IBM WebSphere MQ Advanced Message Security (AMS) é um componente licenciado separadamente do IBM WebSphere MQ que fornece um alto nível de proteção para dados sensíveis que fluem pela rede do IBM WebSphere MQ , enquanto não impacta os aplicativos finais.

Referências relacionadas

Referência de saída de API

Chamadas de Saída do Canal e Estrutura de Dados

Planejando a auditoria

Decida quais dados são necessários para a auditoria e como as informações de auditoria serão capturadas e processadas. Considere como verificar se o sistema está configurado corretamente.

Há vários aspectos para o monitoramento de atividade. Os aspectos que devem ser considerados são frequentemente definidos por requisitos de auditoria, e esses requisitos são geralmente orientados por normas regulamentares, como o HIPAA (Health Insurance Portability and Accountability Act) ou o SOX (Sarbanes-Oxley). O IBM WebSphere MQ fornece recursos destinados a ajudar na conformidade com tais normas.

Considere se você está interessado apenas em exceções ou se está interessado em todo o comportamento do sistema.

Alguns aspectos de auditoria também podem ser considerados como monitoramento operacional; uma distinção para a auditoria é que você está sempre olhando para dados históricos, e não apenas para alertas em tempo real. O monitoramento é coberto na seção Monitoramento e desempenho

Quais dados auditar

Considere quais tipos de dados ou atividade são necessários para auditar, conforme descrito nas seções a seguir:

Mudanças feitas em IBM WebSphere MQ usando as interfaces IBM WebSphere MQ
Configure o IBM WebSphere MQ para emitir os eventos de instrumentação, especificamente os eventos de comando e eventos de configuração.

Mudanças feitas no IBM WebSphere MQ fora de seu controle
Algumas mudanças podem afetar como o IBM WebSphere MQ se comporta, mas não podem ser monitoradas diretamente pelo IBM WebSphere MQ. Exemplos de tais mudanças incluem mudanças nos arquivos de configuração `mqs.ini`, `qm.ini` e `mqclient.ini`, a criação e a exclusão de gerenciadores de filas, instalação de arquivos binários como programas de saída de usuários e alterações de permissões de arquivo. Para monitorar essas atividades, deve-se usar ferramentas em execução no nível do sistema operacional. Diferentes ferramentas estão disponíveis e são apropriadas para diferentes sistemas operacionais. Também é necessário ter logs criados por ferramentas associadas, tais como *sudo*.

Controle operacional de IBM WebSphere MQ
Talvez seja necessário usar as ferramentas do sistema operacional para auditar atividades, como iniciar e parar os gerenciadores de filas. Em alguns casos, IBM WebSphere MQ pode ser configurado para emitir eventos de instrumentação.

Atividade do Aplicativo no IBM WebSphere MQ
Para auditar as ações de aplicativos, por exemplo, abrir filas e enviar e receber mensagens, configure o IBM WebSphere MQ para emitir eventos adequados.

Alertas de intruso
Para auditar tentativas de violação de segurança, configure o sistema para emitir eventos de autorização. Os eventos do canal também podem ser úteis para mostrar a atividade, especialmente se um canal for encerrado inesperadamente.

Planejando a captura, exibição e arquivamento de dados de auditoria

Muitos dos elementos que são necessários são relatados como mensagens do evento do IBM WebSphere MQ. Deve-se escolher ferramentas que podem ser lidas e formatar essas mensagens. Se você estiver interessado em armazenamento e análise de longo prazo, deverá movê-los para um mecanismo de armazenamento auxiliar, como um banco de dados. Se essas mensagens não forem processadas, elas permanecerão na fila de eventos, possivelmente preenchendo a fila. É possível escolher implementar uma ferramenta que executa automaticamente uma ação com base em alguns eventos; por exemplo, para emitir um alerta quando uma falha de segurança ocorre.

Verificando se seu sistema está corretamente configurado

Um conjunto de testes é fornecido com o IBM WebSphere MQ Explorer. Use estes testes para verificar problemas em suas definições de objeto.

Além disso, verifique periodicamente se a configuração do sistema está conforme o esperado. Embora eventos de comando e configuração possam relatar quando algo é mudado, eles também são úteis para fazer dump da configuração e compará-la com uma cópia correta conhecida.

Planejando a segurança por meio da topologia

Esta seção abrange a segurança em situações específicas, nomeadamente para os canais, clusters de gerenciadores de filas, publicação/assinatura e aplicativos multicast, e ao utilizar um firewall.

Consulte os subtópicos a seguir para obter mais informações:

Autorização de canal

Ao enviar ou receber uma mensagem por meio de um canal, você precisa de um ID do usuário que tenha acesso a vários recursos do IBM WebSphere MQ .

Para receber mensagens no tempo de PUT para MCAs, é possível usar o ID de usuário associado ao MCA ou o ID do usuário associado à mensagem.

No tempo CONNECT, é possível mapear o ID do usuário declarado para um usuário alternativo, usando registros de autenticação do canal **CHLAUTH**.

No WebSphere MQ, os canais podem ser protegidos pelo suporte de SSL ou TLS

Os IDs de usuário associados ao envio e recebimento de canais, excluindo o canal emissor em que o atributo MCAUSER não é usado, precisam ter acesso aos seguintes recursos:

- O ID do usuário associado a um canal de envio requer acesso ao gerenciador de filas, a fila de transmissão, a fila de mensagens não entregues e acesso a quaisquer outros recursos que são requeridos por saídas do canal.
- O ID do usuário MCAUSER de um canal receptor precisa da autoridade *+setall*.

  A razão é que o canal receptor tem que criar o MQMD completo, incluindo todos os campos de contexto, usando os dados que ele recebeu do canal emissor remoto.

  O gerenciador de filas, portanto, requer que o usuário que executa esta atividade tenha a autoridade *+setall*. Esta autoridade *+setall* deve ser concedida ao usuário para:

  – Todas as filas que o canal receptor coloca validamente as mensagens.
  – O objeto do gerenciador de filas. Consulte Autorizações para o contexto para obter informações adicionais.
- O ID do usuário MCAUSER de um canal receptor no qual o originador solicitou uma mensagem de relatório COA precisa de autoridade *+passid* na fila de transmissão que retorna a mensagem de relatório. Sem essa autoridade, as mensagens de erro AMQ8077 são registradas.
- Com o ID do usuário associado ao canal receptor, é possível abrir as filas de destino para colocar mensagens nas filas.

  Isso envolve o Message queuing Interface (MQI), portanto, as verificações de controle de acesso adicionais poderão precisar ser feitas se você não estiver usando o WebSphere MQ Object Authority Manager (OAM). É possível especificar se as verificações de autorização são feitas em relação ao ID de usuário associado ao MCA (conforme descrito neste tópico) ou em relação ao ID de usuário associado à mensagem (a partir do campo MQMD UserIdentifier).

  Para os tipos de canais para os quais ele se aplica, o parâmetro **PUTAUT** de uma definição de canal especifica qual ID de usuário é usado para essas verificações.

  – O canal é padronizado para usar a conta do serviço do gerenciador de filas, que terá direitos administrativos completos e não requererá autorizações especiais

    No caso de canais de conexão do servidor, as conexões administrativas são bloqueadas por padrão pelas regras CHLAUTH e requerem fornecimento explícito.

    Canais do tipo receptor, solicitante e receptor de cluster permitem administração local por qualquer gerenciador de filas adjacente, a menos que o administrador execute as etapas para restringir esse acesso.
  – Se você usar um ID do usuário que não possui privilégios administrativos do WebSphere , deverá conceder autoridade dsp e ctrlx para o canal para esse ID do usuário para o canal funcionar. O atributo MCAUSER não é usado para o tipo de canal SDR.
  – Se você usar o ID do usuário associado à mensagem, é provável que o ID do usuário seja de um sistema remoto.

    Esse ID do usuário do sistema remoto deve ser reconhecido pelo sistema de destino. Por exemplo, emita os seguintes comandos:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect +inq +setall
```

```
setmqaut -m QMgrName -t chl -n Profile -g GroupName +dsp +ctrlx
```

em que *Profile* é um canal.

```
setmqaut -m QMgrName -t q -n Profile -g GroupName +put +setall
```

em que *Profile* é uma fila de mensagens não entregues, se configurada.

```
setmqaut -m QMgrName -t q -n Profile -g GroupName +put +setall
```

em que *Profile* é uma lista de filas autorizadas.

Atenção: Tome cuidado ao autorizar um ID do usuário a colocar mensagens na fila de comandos ou outras filas sensíveis do sistema.

O ID de usuário associado ao MCA depende do tipo de MCA. Há dois tipos de MCA:

MCA responsável pela chamada
MCAs que iniciam um canal. Os MCAs responsáveis pela chamada podem ser iniciados como processos individuais, como encadeamentos do inicializador de canais, ou como encadeamentos de um conjunto de processos. O ID do usuário usado é o ID do usuário associado ao processo pai (o inicializador de canais) ou o ID do usuário associado ao processo que inicia o MCA.

MCA respondente
MCAs respondentes são MCAs que são iniciados como resultado de uma solicitação feita por um MCA responsável pela chamada. MCAs respondentes podem ser iniciados como processos individuais, como encadeamentos do listener ou como encadeamentos de um conjunto de processos. O ID do usuário pode ser qualquer um dos tipos a seguir (nessa ordem, de preferência):

1. No APPC, o MCA responsável pela chamada pode indicar o ID de usuário a ser usado para o MCA respondente. Isso é chamado de ID do usuário da rede, e se aplica somente a canais iniciados como processos individuais. Configure o ID do usuário de rede usando o parâmetro USERID da definição de canal.
2. Se o parâmetro **USERID** não for usado, a definição de canal do MCA respondente pode especificar o ID do usuário que o MCA deve usar. Configure o ID do usuário usando o parâmetro **MCAUSER** da definição de canal.
3. Se o ID do usuário não foi definido por um dos métodos anteriores (dois), o ID do usuário do processo que inicia o MCA ou o ID do usuário do processo pai (o listener) é usado.

Conceitos relacionados

“Registros de Autenticação de Canal” na página 40
Para exercer maior controle preciso sobre o acesso concedido à conexão de sistemas em um nível de canal, é possível usar registros de autenticação de canal.

Propriedades de registro de autenticação de canal

Protegendo Definições do Inicializador de Canais

Apenas membros do grupo mqm podem manipular inicializadores de canais.

Os inicializadores de canais do IBM WebSphere MQ não são objetos do IBM WebSphere MQ; o acesso a eles não é controlado pelo OAM. O IBM WebSphere MQ não permite que usuários ou aplicativos manipulem esses objetos, a menos que o ID do usuário seja um membro do grupo mqm. Se você tiver um aplicativo que emita o comando PCF `StartChannelInitiator`, o ID do usuário especificado no descritor de mensagem da mensagem PCF deverá ser um membro do grupo mqm no gerenciador de filas de destino...

Um ID de usuário também deverá ser um membro do grupo mqm na máquina de destino para emitir os comandos MQSC equivalentes por meio do comando PCF Escape ou usando `runmqsc` no modo indireto.

Filas de transmissão
Os gerenciadores de filas colocam mensagens remotas automaticamente em uma fila de transmissão; não é necessária nenhuma autoridade especial para isso.

No entanto, se for necessário colocar uma mensagem diretamente em uma fila de transmissão, isso exigirá autorização especial; consulte Tabela 10 na página 92.

Saídas do canal
Se registros de autenticação de canal não forem adequados, será possível usar saídas do canal para segurança incluída. Uma saída de segurança forma uma conexão segura entre dois programas de saída de segurança. Um programa é para o agente do canal de mensagens de envio (MCA) e o outro para o MCA de recebimento.

Consulte “Programas de Saída de Canal” na página 66 para obter mais informações sobre saídas do canal.

Protegendo canais com SSL
O suporte SSL no IBM WebSphere MQ usa o objeto de informações de autenticação do gerenciador de filas e vários comandos do MQSC Também se deve considerar o uso de certificados digitais.

Comandos e Atributos para Suporte SSL

O protocolo Secure Sockets Layer (SSL) fornece segurança de canal, com proteção contra espionagem, violação e representação. O suporte do IBM WebSphere MQ para SSL permite especificar, na definição de canal, que um determinado canal usa a segurança SSL. Também é possível especificar detalhes do tipo de segurança desejado, como o algoritmo de criptografia que você deseja usar.

Os comandos MQSC a seguir suportam SSL:

ALTER AUTHINFO
Modifica os atributos de um objeto de informações sobre autenticação.

DEFINE AUTHINFO
Cria um objeto de informações sobre autenticação.

DELETE AUTHINFO
Exclui um objeto de informações sobre autenticação.

DISPLAY AUTHINFO
Exibe os atributos para um objeto de informações sobre autenticação específico.

Os seguintes parâmetros do gerenciador de filas suportam SSL:

SSLCRLNL
O atributo SSLCRLNL especifica uma lista de nomes de objetos de informações sobre autenticação que são usados para fornecer locais de revogação de certificado para permitir verificação aprimorada de certificados TLS/SSL.

SSLCRYP
Em sistemas Windows, UNIX and Linux , configura o atributo do gerenciador de filas SSLCryptoHardware Esse atributo é o nome da sequência de parâmetros que pode ser usada para configurar o hardware criptográfico que você tem no sistema.

SSLEV
Determina se uma mensagem de evento SSL será relatada se um canal usando SSL falhar ao estabelecer uma conexão SSL.

SSLFIPS
Especifica se apenas algoritmos certificados por FIPS devem ser usados se a criptografia for executada no IBM WebSphere MQ, em vez de no hardware de criptografia. Se o hardware de criptografia for configurado, os módulos de criptografia fornecidos pelo produto de hardware são usados, e estes podem ser certificados por FIPS em um nível específico. Isto depende do produto de hardware em uso.

SSLKEYR
Em sistemas Windows,UNIX and Linux , associa um repositório de chaves a um gerenciador de filas O banco de dados de chaves fica retido em um banco de dados de chaves *GSKit*. (O IBM Global Security Kit (GSKit) permite usar a segurança SSL em sistemas Windows,UNIX and Linux .)

SSLRKEYC
O número de bytes a serem enviados e recebidos em uma conversa SSL antes da chave secreta ser renegociada. O número de bytes inclui informações de controle enviadas pelo MCA.

Os parâmetros de canal a seguir suportam SSL:

SSLCAUTH
Define se o IBM WebSphere MQ requer e valida um certificado do cliente SSL.

SSLCIPH
Especifica a segurança da criptografia e a função (CipherSpec), por exemplo, NULL_MD5 ou RC4_MD5_US. O CipherSpec deve corresponder em ambas as extremidades do canal.

SSLPEER
Especifica o nome distinto (identificador exclusivo) de parceiros permitidos.

Esta seção descreve os comandos `setmqaut`, `dspmqaut`, `dmpmqaut`, `rcrmqobj`, `rcdmqimge` `dspmqfls` para suportar o objeto de informações sobre autenticação.. Ele também descreve o comando iKeycmd para gerenciar certificados em sistemas UNIX and Linux e a ferramenta runmqakm para gerenciar certificados nos sistemas UNIX, Linux e Windows . Consulte as seções a seguir:

- setmqaut
- dspmqaut
- dmpmqaut
- rcrmqobj
- rcdmqimg
- dspmqfls
- Gerenciando chaves e certificados

Para obter uma visão geral da segurança do canal usando SSL, consulte

- “IBM WebSphere MQ Suporte para SSL e TLS” na página 23

Para obter detalhes de comandos MQSC associados a SSL, veja

- ALTER AUTHINFO
- DEFINE AUTHINFO
- DELETE AUTHINFO
- DISPLAY AUTHINFO

Para obter detalhes sobre comandos de PCF associados com SSL, consulte

- Mudar, copiar e criar objeto de informações sobre autenticação
- Excluir Objeto de Informações sobre Autenticação
- Investigar Objeto de Informações sobre Autenticação

Certificados autoassinados e certificados assinados por CA

É importante planejar o uso de certificados digitais quando se está desenvolvendo e testando seu aplicativo, e para seu uso em produção. É possível usar os certificados assinados por CA ou os certificados autoassinados, dependendo do uso de seus gerenciadores de filas e aplicativos clientes.

Certificados assinados pelo CA
Para sistemas de produção, obtenha os certificados a partir de uma autoridade de certificação (CA) confiável. Ao se obter um certificado a partir de uma CA externa, você paga pelo serviço.

Certificados autoassinados

Enquanto você está desenvolvendo seu aplicativo, será possível usar certificados autoassinados ou certificados emitidos por uma autoridade de certificação local, dependendo da plataforma:

Nos sistemas Windows, UNIXe Linux , é possível usar certificados autoassinados.. Consulte a seção “Criando um certificado pessoal auto-assinado em sistemas UNIX, Linux, and Windows” na página 124 para obter instruções.

Os certificados autoassinados não são adequados para uso de produção, pelas seguintes razões:

- Certificados autoassinados não podem ser revogados, o que pode permitir que um invasor realize spoof em uma identidade após uma chave privada ter sido comprometida. CAs podem revogar um certificado comprometido, evitando seu uso adicional. O uso de certificados assinados por CA são, portanto, mais seguros em um ambiente de produção, embora certificados autoassinados sejam mais convenientes em um sistema de teste.
- Os certificados autoassinados nunca expirarão. Isso é conveniente e seguro em um ambiente de teste, mas em um ambiente de produção, isso os deixa abertos a eventuais violações de segurança. O risco é ainda composto pelo fato de os certificados autoassinados não poderem ser revogados.
- Um certificado autoassinado é usado como um certificado pessoal e como um certificado de autoridade de certificação raiz (ou âncora de confiança). Um usuário com um certificado pessoal autoassinado pode ser capaz de usá-lo para assinar outros certificados pessoais. Em geral, isso não é verdadeiro para certificados pessoais emitidos por uma autoridade de certificação, e representa uma exposição significativa.

CipherSpecs e certificados digitais

Somente um subconjunto dos CipherSpecs suportados pode ser usado com todos os tipos suportados de certificado digital. Portanto, é necessário escolher um CipherSpec apropriado para o seu certificado digital. Da mesma forma, se a política de segurança de sua organização requer que um CipherSpec específico seja usado; então, deve-se obter um certificado digital adequado.

Para obter mais informações sobre o relacionamento entre CipherSpecs e certificados digitais, consulte a “Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ” na página 35

Políticas de Validação de Certificado

O padrão IETF RFC 5280 especifica uma série de regras de validação de certificado que o software de aplicativo compatível deve implementar para impedir ataques de personificação. Um conjunto de regras de validação de certificado é conhecido como uma política de validação de certificado. Para obter mais informações sobre as políticas de validação de certificado no WebSphere MQ, consulte “Políticas de validação de certificado no IBM WebSphere MQ” na página 34

serviços de segurança do SNA LU 6.2

O SNA LU 6.2 oferece criptografia em nível de sessão, autenticação em nível de sessão e autenticação em nível de conversa.

Nota: Esta coleção de tópicos supõe que você tenha um entendimento básico de Systems Network Architecture (SNA). A outra documentação referida nesta seção contém uma breve introdução aos conceitos e terminologia relevantes. Se precisar de uma introdução técnica mais abrangente ao SNA, consulte *Systems Network Architecture Technical Overview*, GC30-3073.

O SNA LU 6.2 oferece três serviços de segurança:

- Criptografia em nível de sessão
- Autenticação em nível de sessão
- Autenticação em nível de conversação

Para criptografia em nível de sessão e autenticação em nível de sessão, o SNA utiliza o algoritmo do *DES (Data Encryption Standard)*. O algoritmo do DES é um algoritmo de cifra de bloco, que utiliza uma chave simétrica para criptografar e decriptografar dados. O bloco e a chave têm oito bytes de comprimento.

Criptografia em nível de sessão
A *Criptografia em nível de sessão* criptografa e decriptografa dados de sessão utilizando o algoritmo do DES. Ela pode portanto, ser utilizada para fornecer um serviço de confidencialidade em nível de link em canais do SNA LU 6.2.

LUs (Logical units) podem fornecer criptografia de dados obrigatória (ou necessária), criptografia de dados seletiva ou nenhuma criptografia de dados.

Em uma *sessão criptográfica obrigatória*, uma LU criptografa todas as unidades de pedido de dados de transmissão e decriptografa todas as unidades de pedido de dados de recepção.

Em uma *sessão criptográfica seletiva*, uma LU criptografa apenas as unidades de pedido de dados especificadas pelo TP (transaction program) de envio. A LU de envio indica que os dados são criptografados definindo um indicador no cabeçalho de pedido. Verificando esse indicador, a LU de recebimento pode definir quais unidades de pedido serão decriptografadas antes de transferi-las para o TP de recebimento.

Em uma rede SNA, WebSphere MQ MCAs são programas de transação. Os MCAs não solicitam criptografia para os dados que enviam. Portanto, a criptografia de dados seletiva não é uma opção; apenas a criptografia de dados obrigatória ou nenhuma criptografia de dados é possível em uma sessão.

Para obter informações sobre como implementar a criptografia de dados obrigatória, consulte a documentação para seu subsistema SNA. Consulte a mesma documentação para obter informações sobre formas mais fortes de criptografia que podem estar disponíveis para uso em sua plataforma, como criptografia Triple DES de 24 bytes no z/OS.

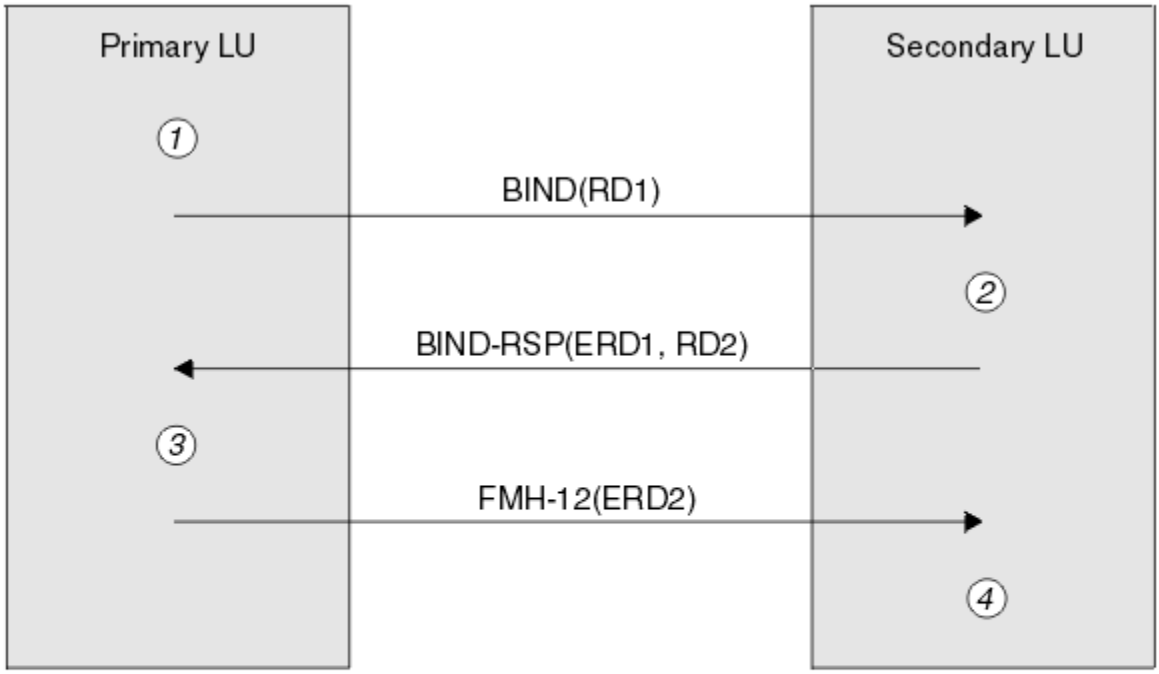
Para obter mais informações gerais sobre criptografia em nível de sessão, consulte *Systems Network Architecture LU 6.2 Reference: Peer Protocols* , SC31-6808.

Autenticação em nível de sessão
A *Autenticação em nível de sessão* é um protocolo de segurança em nível de sessão que permite que duas LUs autentiquem uma à outra enquanto ativam a sessão. Também conhecida como *verificação de LU-LU*.

Como uma LU é efetivamente o "gateway" para um sistema a partir da rede, você pode considerar esse nível de autenticação como suficiente em certas circunstâncias. Por exemplo, se seu gerenciador de filas precisar trocar mensagens com um gerenciador de filas remoto sendo executado em um ambiente controlado e confiável, você pode estar preparado para confiar nas identidades dos componentes restantes do sistema remoto após a autenticação da LU.

A autenticação em nível de sessão é conseguida por cada LU verificando a senha de seu parceiro. A senha é denominada uma *senha de LU-LU* porque é estabelecida uma senha entre cada par de LUs. A maneira que uma senha de LU-LU é estabelecida depende da implementação e está fora do escopo do SNA.

Figura 10 na página 77 ilustra os fluxos para autenticação em nível de sessão.



Legend:

BIND	=	BIND request unit
BIND-RSP	=	BIND response unit
ERD	=	Encrypted random data
FMH-12	=	Function Management Header 12
RD	=	Random data

Figura 10. Fluxos para autenticação em nível de sessão

O protocolo para autenticação em nível de sessão é o seguinte. Os números no procedimento correspondem aos números em Figura 10 na página 77.

1. A LU primária gera um valor de dados aleatório (RD1) e o envia para a LU secundária no pedido BIND.
2. Quando a LU secundária recebe o pedido BIND com os dados aleatórios, ela criptografa os dados utilizando o algoritmo do DES com sua cópia da senha de LU-LU como a chave. A LU secundária, então, gera um segundo valor de dados aleatórios (RD2) e envia-o com os dados criptografados (ERD1) para a LU primária na resposta BIND.
3. Quando a LU primária recebe a resposta BIND, ela calcula sua própria versão dos dados criptografados a partir dos dados aleatórios gerados originalmente. Ela o faz utilizando o algoritmo do DES com sua cópia da senha de LU-LU como a chave. Então ela compara sua versão com os dados criptografados recebidos na resposta BIND. Se os dois valores forem iguais, a LU primária saberá que a LU secundária tem a mesma senha que ela e a LU secundária será autenticada. Se os dois valores não corresponderem, a LU primária encerrará a sessão.

   A LU primária criptografa os dados aleatórios recebidos na resposta BIND e envia os dados criptografados (ERD2) para a LU secundária em um FMH-12 (Function Management Header 12).
4. Quando a LU secundária recebe o FMH-12, ela calcula sua própria versão dos dados criptografados a partir dos dados aleatórios gerados por ela. Então ela compara sua versão com os dados criptografados recebidos no FMH-12. Se os dois valores forem iguais, a LU primária será autenticada. Se os dois valores não corresponderem, a LU secundária encerrará a sessão.

Em uma versão aperfeiçoada do protocolo, que fornece melhor proteção contra ataques humano intermediários, a LU secundária calcula um DES MAC (Message Authentication Code) a partir de RD1,

RD2 e o nome completo da LU secundária, utilizando sua cópia da senha de LU-LU como a chave. A LU secundária envia o MAC para a LU primária na resposta BIND em vez de ERD1.

A LU primária autentica a LU secundária calculando sua própria versão do MAC, a qual ela compara com o MAC recebido na resposta BIND. Em seguida a LU primária calcula um segundo MAC a partir de RD1 e RD2, e envia o MAC para a LU secundária no FMH-12 em vez do ERD2.

A LU secundária autentica a LU primária calculando sua própria versão do segundo MAC, a qual ela compara com o MAC recebido no FMH-12.

Para obter informações sobre como configurar a autenticação em nível de sessão, consulte a documentação para seu subsistema SNA. Para obter mais informações gerais sobre autenticação de nível de sessão, consulte *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808.

Autenticação em nível de conversação
Quando um TP local tenta alocar uma conversação com um TP parceiro, a LU local envia um pedido de anexo para a LU parceira, pedindo que anexe o TP parceiro. Em certas circunstâncias, o pedido de anexo pode conter informações de segurança que a LU parceira pode utilizar para autenticar o TP local. Isso é conhecido como *autenticação em nível de conversação* ou *verificação do usuário final*.

Os tópicos a seguir descrevem como o IBM WebSphere MQ fornece suporte para autenticação em nível de conversação.

Para obter mais informações sobre autenticação em nível de conversação, consulte *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808. Para obter informações específicas para o z/OS, consulte *z/OS MVS Planning: APPC/MVS Management*, SA22-7599.

Para obter mais informações sobre o CPI-C, consulte *Common Programming Interface Communications CPI-C Specification*, SC31-6180. Para obter mais informações sobre APPC/MVS TP Conversation Callable Services, consulte *z/OS MVS Programming: Writing Transaction Programs for APPC/MVS*, SA22-7621.

Suporte para autenticação de nível de conversa em IBM WebSphere MQ nos sistemas UNIX e Windows
Use este tópico para obter uma visão geral de como a autenticação em nível de conversa funciona, no UNIX, Linux, and Windows.

O suporte para autenticação de nível de conversa em IBM WebSphere MQ para WebSphere MQ em sistemas UNIX e WebSphere MQ para Windows é ilustrado em Figura 11 na página 79 Os números no diagrama correspondem aos números na descrição a seguir.

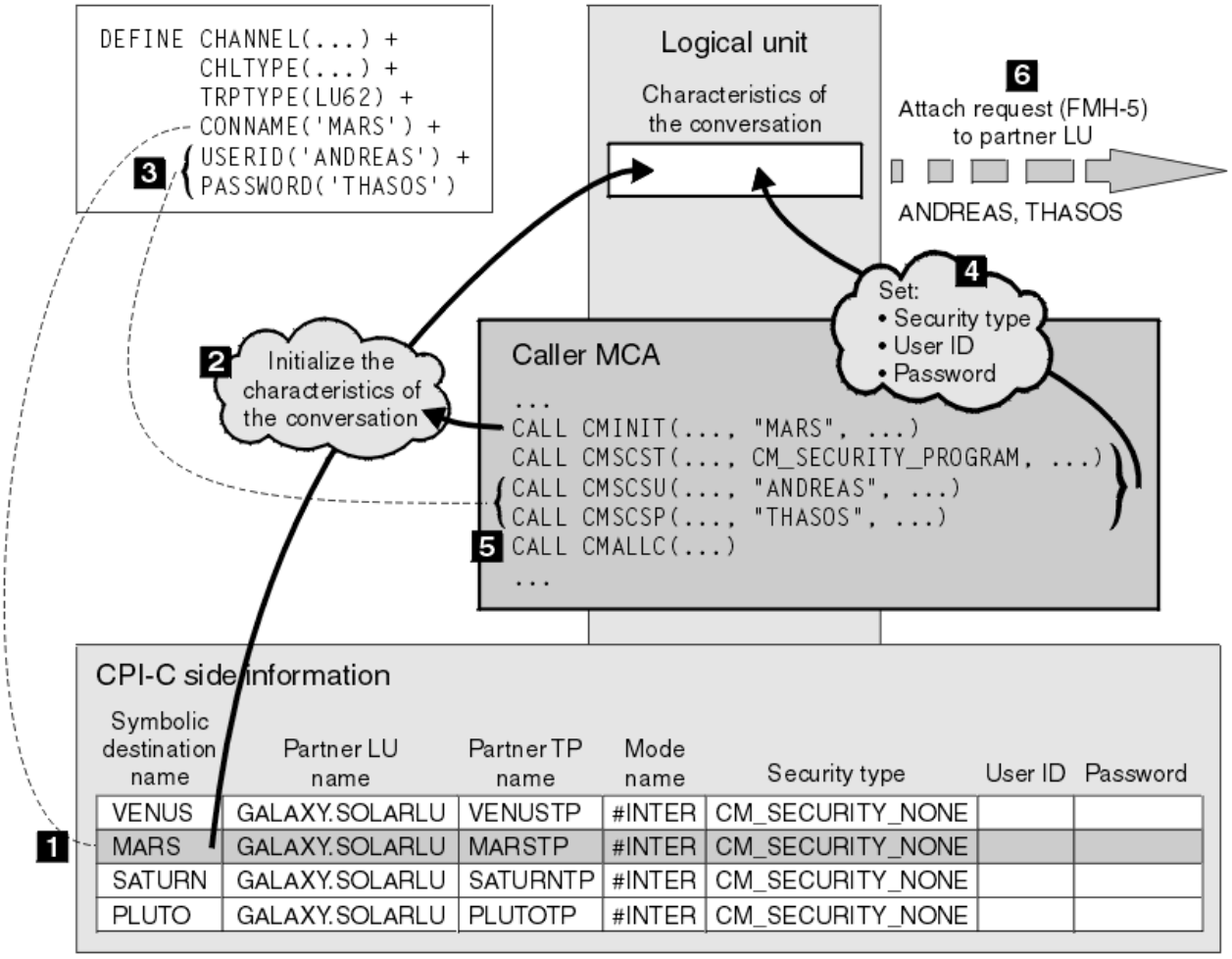


Figura 11. Suporte do WebSphere MQ para autenticação no nível de conversa

Em sistemas IBM i, UNIX e sistemas Windows , um MCA usa chamadas CPI-C (Common Programming Interface Communications) para se comunicar com um MCA parceiro em uma rede SNA. Na definição de canal na extremidade do responsável pela chamada de um canal, o valor do parâmetro CONNAME é um nome de destino simbólico, que identifica uma entrada de informações secundárias de CPI-C (1). Essa entrada especifica:

- O nome da LU parceira
- O nome do TP parceiro, que é um MCA receptor da chamada
- O nome do modo a ser utilizado para a conversação

Uma entrada de informações secundárias também pode especificar as seguintes informações de segurança:

- Um tipo de segurança.

  Os tipos de segurança geralmente implementados são CM_SECURITY_NONE, CM_SECURITY_PROGRAM e CM_SECURITY_SAME, mas outros são definidos na especificação de CPI-C.
- Um ID do usuário.
- Uma senha.

Um MCA originador da chamada se prepara para alocar uma conversação com um MCA receptor da chamada emitindo a chamada de CPI-C CMINIT, utilizando o valor de CONNAME como um dos parâmetros na chamada. A chamada CMINIT identifica, para o benefício da LU local, a entrada de informações secundárias que o MCA pretende utilizar para a conversação. A LU local usa os valores nesta entrada para inicializar as características da conversa (2).

O MCA do responsável pela chamada verifica os valores dos parâmetros USERID e PASSWORD na definição de canal (3). Se USERID for definido, o MCA do responsável pela chamada emite as seguintes chamadas de CPI-C (4):

- CMSCST, para definir o tipo de segurança para a conversação para CM_SECURITY_PROGRAM.
- CMSCSU, para definir o ID de usuário para a conversação para o valor de USERID.
- CMSCSP, para definir a senha para a conversação para o valor de PASSWORD. CMSCSP não é chamado a menos que PASSWORD seja definido.

O tipo de segurança, ID de usuário e senha definidos por essas chamadas substituem quaisquer valores adquiridos anteriormente da entrada de informações secundárias.

O MCA do responsável pela chamada emite a chamada de CPI-C CMALLC para alocar a conversa (5). Em resposta a essa chamada, a LU local envia uma solicitação de conexão (Function Management Header 5 ou FMH-5) para a LU do parceiro (6).

Se a LU parceira aceitar um ID de usuário e uma senha, os valores de USERID e de PASSWORD serão incluídos no pedido de anexo. Se a LU parceira não aceitar um ID de usuário e uma senha, os valores não serão incluídos no pedido de anexo. A LU local descobre se a LU parceira aceitará um ID de usuário e uma senha como parte de uma troca de informações quando as LUs se ligarem para formar uma sessão.

Em uma versão posterior do pedido de anexo, um substituto de senha pode fluir entre as LUs em vez de uma senha propriamente dita.Um substituto de senha é um MAC (Message Authentication Code) ou uma compilação de mensagens SHA-1, formados a partir da senha. Substitutos de senha podem ser utilizados apenas se ambas as LUs os suportarem.

Quando a LU parceira recebe um pedido de anexo de entrada contendo um ID de usuário e uma senha, ela pode utilizar o ID de usuário e a senha para os objetivos de identificação e de autenticação. Referindo-se às listas de controle de acesso, a LU parceira também pode determinar se o ID de usuário tem autoridade para alocar uma conversação e anexar o MCA receptor da mensagem.

Além disso, o MCA receptor da mensagem pode ser executado sob o ID de usuário incluído no pedido de anexo.Nesse caso, o ID de usuário se torna o ID de usuário padrão para o MCA receptor da chamada e é utilizado para verificações de autoridade quando o MCA tenta se conectar ao gerenciador de filas. Ele também pode ser utilizado para verificações de autoridade subseqüentemente quando o MCA tenta acessar os recursos do gerenciador de filas.

A forma como um ID de usuário e uma senha em um pedido de anexo podem ser utilizados para identificação, autenticação e controle de acesso depende da implementação. Para obter informações específicas para seu subsistema SNA, consulte a documentação apropriada.

Se USERID não for definido, o MCA originador da chamada não chamará CMSCST, CMSCSU e CMSCSP. Nesse caso, as informações de segurança que circulam em um pedido de anexo são determinadas unicamente pelo que for especificado na entrada de informações secundárias e o que a LU parceira irá aceitar.

Segurança para Clusters de Gerenciadores de Filas

Embora o uso dos clusters de gerenciadores de filas possa ser conveniente, você deve prestar muita atenção à sua segurança.

Um *cluster de gerenciadores de filas* é uma rede de gerenciadores de filas que estão associados logicamente de alguma maneira. Um gerenciador de filas que seja um membro de um cluster é chamado um *gerenciador de filas de cluster*.

Uma fila que pertença a um gerenciador de filas de cluster pode ser tornada conhecida a outros gerenciadores de filas no cluster. Uma fila assim é chamada uma *fila de cluster*. Qualquer gerenciador de filas em um cluster pode enviar mensagens para filas do cluster sem precisar de nenhum dos seguintes:

- Uma definição explícita de fila remota para cada fila do cluster
- Canais explicitamente definidos para e de cada gerenciador de filas remotas
- Uma fila de transmissão separada para cada canal de transmissão

É possível criar um cluster no qual dois ou mais gerenciadores de filas são clones. Isso significa que eles possuem ocorrências das mesmas filas locais, incluindo quaisquer filas locais declaradas como filas de cluster, e podem suportar ocorrências dos mesmos aplicativos do servidor.

Quando um aplicativo conectado a um gerenciador de filas do cluster envia uma mensagem a uma fila de clusters que tem uma instância em cada um dos gerenciadores de filas clonados, o IBM WebSphere MQ decide para qual gerenciador de filas enviá-la. Quando muitos aplicativos enviam mensagens para a fila de clusters, o WebSphere MQ equilibra a carga de trabalho em cada um dos gerenciadores de fila que possuem uma instância da fila Se um dos sistemas que hospeda um gerenciador de filas clonado falhar, o WebSphere MQ continuará a equilibrar a carga de trabalho nos gerenciadores de filas restantes até que o sistema que falhou seja reiniciado.

Se você estiver utilizando clusters de gerenciadores de filas, precisará levar em consideração as seguintes questões de segurança:

- Permitir que somente gerenciadores de filas selecionados enviem mensagens a seu gerenciador de filas
- Permitir que somente usuários selecionados de um gerenciador de filas remoto enviem mensagens a uma fila no seu gerenciador de filas
- Permitir que aplicativos conectados a seu gerenciador de filas enviem mensagens somente a filas remotas selecionadas

Essas considerações são relevantes mesmo que você não esteja utilizando clusters, mas se tornam mais importantes se estiverem sendo utilizados clusters.

Se um aplicativo puder enviar mensagens a uma fila de cluster, ele poderá enviar mensagens a qualquer outra fila do cluster sem precisar de definições adicionais de filas remotas, filas de transmissão ou canais. Portanto, torna-se mais importante considerar se é preciso restringir o acesso às filas do cluster em seu gerenciador de filas, e restringir as filas do cluster às quais os aplicativos podem enviar mensagens.

Existem algumas considerações de segurança adicionais que são relevantes somente se você estiver utilizando clusters de gerenciadores de filas:

- Permitir que somente gerenciadores de filas selecionados se unam a um cluster
- Forçando Gerenciadores de Filas Indesejáveis a Sair de um Cluster

Para obter mais informações sobre todas essas considerações, consulte Mantendo os Clusters Seguros.

Tarefas relacionadas

“Impedindo que gerenciadores de filas recebam mensagens” na página 253
É possível evitar que um gerenciador de filas do cluster receba mensagens que ele não está autorizado a receber, usando programas de saída.

Segurança para Publicação / Assinatura do IBM WebSphere MQ

Há considerações de segurança adicionais se você estiver usando o publicar/assinar do IBM WebSphere MQ.

Em um sistema de publicação/assinatura, há dois tipos de aplicativo: publicador e assinante. Os *publicadores* fornecem informações no formato de mensagens do IBM WebSphere MQ. Quando um publicador publica uma mensagem, ele especifica um *tópico*, que identifica o assunto das informações dentro da mensagem.

Assinantes são os consumidores das informações que são publicadas. Um assinante especifica os tópicos nos quais está interessado inscrevendo-se neles.

O *gerenciador de filas* é um aplicativo fornecido com o Publicar/assinar do IBM WebSphere MQ. Ele recebe as mensagens publicadas dos publicadores e pedidos de assinatura dos assinantes, e encaminha as mensagens publicadas aos assinantes. São enviadas a um assinante somente as mensagens sobre os tópicos que ele assinou.

Para obter mais informações, consulte Segurança da Publicação/Assinatura.

Segurança de multicast

Use estas informações para entender por que os processos de segurança podem ser necessários com o IBM WebSphere MQ Multicast.

O IBM WebSphere MQ Multicast não tem segurança integrada. As verificações de segurança são manipuladas no gerenciador de filas no tempo de MQOPEN, e a configuração do campo MQMD é manipulada pelo cliente. Alguns aplicativos na rede podem não ser aplicativos IBM WebSphere MQ (por exemplo, aplicativos LLM, consulte Interoperabilidade multicast com WebSphere MQ Low Latency Messaging para obter mais informações), portanto, pode ser necessário implementar seus próprios procedimentos de segurança porque os aplicativos de recebimento não podem ter certeza da validade dos campos de contexto.

Há três processos de segurança a serem considerados:

Controle de Acesso

O controle de acesso em IBM WebSphere MQ é baseado nos IDs do usuário. Para obter informações adicionais sobre este assunto, consulte “Controle de acesso para clientes” na página 58.

Segurança de rede

Uma rede isolada pode ser uma opção de segurança viável para evitar mensagens falsas. É possível para um aplicativo no endereço de grupo multicast publicar mensagens maliciosas usando as funções de comunicação nativa, que são indistinguíveis a partir das mensagens do MQ por virem de um aplicativo no mesmo endereço de grupo multicast.

Também é possível para um cliente no endereço de grupo multicast receber mensagens que foram destinadas a outros clientes no mesmo endereço de grupo multicast.

Isolar a rede multicast assegura que apenas clientes e aplicativos válidos possuam acesso. Esta precaução de segurança pode impedir as mensagens maliciosas de entrar e as informação confidenciais de sair.

Para obter informações sobre endereços de rede de grupo multicast, consulte: Configurando a rede apropriada para tráfego multicast

Assinaturas Digitais

Uma assinatura digital é formada pela criptografia de uma representação de uma mensagem. A criptografia utiliza a chave privada do assinante e, para eficiência, geralmente opera em uma compilação de mensagens, ao invés de na mensagem em si. Assinar digitalmente uma mensagem antes de um MQPUT é uma boa precaução de segurança, mas esse processo pode ter um efeito negativo no desempenho se houver um grande volume de mensagens.

As assinaturas digitais variam com os dados que estão sendo assinados. Se duas mensagens diferentes forem assinadas digitalmente pela mesma entidade, as duas assinaturas diferirão, mas ambas poderão ser verificadas com a mesma chave pública, ou seja, a chave pública da entidade que assinou as mensagens.

Conforme mencionado anteriormente nesta seção, pode ser possível para um aplicativo no endereço de grupo multicast publicar mensagens maliciosas usando funções de comunicação nativa, que são indistinguíveis a partir de mensagens do MQ. As assinaturas digitais fornecem prova de origem, e somente o emissor conhece a chave privada, que fornece forte evidência de que o emissor é o originador da mensagem.

Para obter informações adicionais sobre este assunto, consulte “Conceitos criptográficos” na página 7.

Firewalls e intermediário da Internet

Você normalmente usa um firewall para evitar o acesso a partir de endereços IP hostis, por exemplo, em um ataque de negação de serviço. No entanto, você pode precisar bloquear temporariamente endereços IP dentro do IBM WebSphere MQ, talvez enquanto aguarda um administrador de segurança atualizar as regras de firewall.

Para bloquear um ou mais endereços IP, crie um registro de autenticação de canal do tipo ADDRESSMAP ou BLOCKADDR. Para obter mais informações, consulte “Bloqueando Endereços IP Específicos” na página 185.

Segurança para o intermediário de Internet do IBM WebSphere MQ

O intermediário de Internet pode simplificar a comunicação através de um firewall, mas isso tem implicações de segurança.

IBM WebSphere MQ passagem da Internet é uma extensão do produto base IBM WebSphere MQ fornecida em SupportPac MS81.

O WebSphere MQ intermediário da Internet permite que dois gerenciadores de filas troquem mensagens ou um aplicativo cliente WebSphere MQ se conecte a um gerenciador de filas pela Internet sem requerer uma conexão TCP/IP direta. Isso é útil se um firewall proibir uma conexão TCP/IP direta entre dois sistemas. Isso torna a passagem do protocolo de canal do WebSphere MQ flui para dentro e para fora de um firewall mais simples e mais gerenciável, ajustando os fluxos dentro de HTTP ou agindo como um proxy. Usando o SSL (Secure Sockets Layer), ela também pode ser usada para criptografar e decriptografar mensagens que são enviadas através da Internet.

Quando seu sistema WebSphere MQ se comunicar com o IPT, a menos que você esteja usando SSLProxyMode no IPT, assegure-se de que o CipherSpec usado pelo WebSphere MQ corresponda ao CipherSuite usado pelo IPT:

- Quando o IPT está agindo como o servidor SSL ou TLS e o WebSphere MQ está se conectando como o cliente SSL ou TLS, o CipherSpec usado pelo WebSphere MQ deve corresponder a um CipherSuite ativado no conjunto de chaves IPT relevante.
- Quando o IPT está agindo como o cliente SSL ou TLS e está se conectando a um servidor SSL ou TLS do WebSphere MQ , o IPT CipherSuite deve corresponder ao CipherSpec definido no canal WebSphere MQ de recebimento.

Se você migrar do IPT para o suporte SSL e TLS do WebSphere MQ integrado, transfira os certificados digitais do IPT Usando iKeyman.

Para obter mais informações, consulte WebSphere MQ Internet Pass-Thru (SupportPac MS81).

Configurar a segurança

Esta coleção de tópicos contém informações específicas para sistemas operacionais diferentes e para a utilização de clientes.

Configurando a segurança em sistemas UNIX, Linux, and Windows

Considerações de segurança específicas para sistemas UNIX, Linux, and Windows .

Gerenciadores de filas do IBM WebSphere MQ transferem informações que são potencialmente de valor, portanto, é necessário usar um sistema de autoridade para assegurar que usuários não autorizados não possam acessar seus gerenciadores de filas. Considere os seguintes tipos de controles de segurança:

Quem pode administrar o IBM WebSphere MQ
É possível definir o conjunto de usuários que pode emitir comandos para administrar o IBM WebSphere MQ.

Quem pode usar objetos IBM WebSphere MQ
É possível definir quais usuários (geralmente aplicativos) podem usar chamadas MQI e comando de PCF para fazer o seguinte:

- Quem pode se conectar a um gerenciador de filas.
- Quem pode acessar objetos (filas, definições de processos, listas de nomes, canais, canais de conexão do cliente, listeners, serviços e objetos de informações sobre autenticação) e que tipo de acesso eles têm a esses objetos.
- Quem pode acessar as mensagens do IBM WebSphere MQ

- Quem pode acessar as informações de contexto associadas a uma mensagem.

Segurança de canal
É necessário assegurar que os canais usados para enviar mensagens para sistemas remotos possam acessar os recursos necessários.

É possível usar recursos de operação padrão para conceder acesso a bibliotecas de programas, bibliotecas de link do MQI e comandos. No entanto, o diretório que contém as filas e outros dados do gerenciador de filas é privado ao IBM WebSphere MQ; não use comandos do sistema operacional padrão para conceder ou revogar autorizações para recursos do MQI.

Conectando ao IBM WebSphere MQ usando Serviços de Terminal

O direito de usuário **Create global objects** pode causar problemas se você estiver usando Serviços de Terminal.

Se você estiver se conectando a um sistema Windows usando os Serviços de Terminal e tiver problemas para criar ou iniciar um gerenciador de filas, isso pode ser devido ao direito do usuário, **Create global objects**, nas versões recentes de Windows

O direito de usuário do **Create global objects** limita os usuários autorizados a criar objetos no namespace global Para que um aplicativo crie um objeto global, ele deve estar em execução no namespace global ou o usuário no qual o aplicativo está em execução deve ter o direito de usuário **Create global objects** aplicado a ele.

Os administradores têm o direito de usuário **Create global objects** aplicado por padrão, portanto, um administrador pode criar e iniciar gerenciadores de filas quando conectados usando Serviços de Terminal sem alterar os direitos de usuário.

Se os vários métodos de administração do WebSphere MQ não funcionarem ao usar serviços de terminal, tente configurar o direito do usuário **Create global objects** :

1. Abra o painel Ferramentas Administrativas:

   Windows 2003 e Windows XP
   Acesse esse painel usando o **Painel de controle** > **Ferramentas Administrativas**

   Windows Vista e Windows Server 2008
   Acesse esse painel usando **Painel de controle** > **Sistema e manutenção** > **Ferramentas administrativas**.
2. Clique duas vezes em **Política de Segurança Local**.
3. Expanda `Local Policies`..
4. Clique em `User Rights Assignment`.
5. Inclua o novo usuário ou grupo na política do **Create global objects**

Criando e Gerenciando Grupos no Windows

Estas instruções o levam pelo processo de administração de grupos em uma estação de trabalho ou máquina servidor membro.

Para controladores de domínio, usuários e grupos são administrados por meio do Active Directory. Para obter mais detalhes sobre como usar o Active Directory, consulte as instruções apropriadas do sistema operacional.

As mudanças feitas na associação ao grupo de um principal não serão reconhecidas até que o gerenciador de filas seja reiniciado, ou que você emita o comando MQSC REFRESH SECURITY (ou o PCF equivalente).

Use o painel Gerenciamento de Computadores para trabalhar com usuário e grupos. Todas as mudanças feitas para o usuário atual com logon efetuado podem não ser efetivas até que o usuário efetue login novamente.

Windows 2003 e Windows XP
Acesse esse painel usando o **Painel de Controle** > **Ferramentas Administrativas** > **Gerenciamento do Computador**

Windows Vista e Windows Server 2008
Acesse esse painel usando **Painel de controle** > **Sistema e manutenção** > **Ferramentas administrativas** > **Gerenciamento do computador**.

Windows 7
Acesse esse painel usando **Ferramentas administrativas** > **Gerenciamento de computadores**

Criando um grupo no Windows

Crie um grupo usando o painel de controle.

Procedimento

1. Abra o painel de controle
2. Dê um clique duplo em **Ferramentas Administrativas**.
   O painel Ferramentas Administrativas é aberto.
3. Dê um clique duplo em **Gerenciamento de Computadores**.
   O painel Gerenciamento de Computadores é aberto.
4. Expanda **Local Users and Groups**.
5. Clique com o botão direito do mouse em **Grupos** e selecione **Novo Grupo...**.
   O painel Novo Grupo é exibido.
6. Digite um nome apropriado no campo de nome Grupo e, em seguida, clique em **Criar**.
7. Clique em **Fechar**.

Incluindo um Usuário em um Grupo no Windows

Inclua um usuário em um grupo usando o painel de controle.

Procedimento

1. Abra o painel de controle
2. Dê um clique duplo em **Ferramentas Administrativas**.
   O painel Ferramentas Administrativas é aberto.
3. Dê um clique duplo em **Gerenciamento de Computadores**.
   O painel Gerenciamento de Computadores é aberto.
4. No painel Gerenciamento de Computadores, expanda **Usuários e Grupos Locais**.
5. Selecione **Usuários**
6. Clique duas vezes no usuário que você deseja incluir em um grupo.
   O painel de propriedades do usuário é exibido.
7. Selecione a guia **Membro de**.
8. Selecione o grupo no qual você deseja incluir o usuário. Se o grupo desejado não estiver visível:
   a) Clique em **Incluir...**.
      O painel Selecionar Grupos é exibido.
   b) Clique em **Locais...**.
      O painel Locais é exibido.
   c) Selecione o local do grupo em que você deseja incluir o usuário na lista e clique em **OK**.
   d) Digite o nome do grupo no campo fornecido.

      Como alternativa, clique em **Avançado ...** .. e, em seguida, **Localizar Agora** para listar os grupos disponíveis no local atualmente selecionado Daqui, selecione o grupo em que deseja incluir o usuário e clique em **OK**.

e) Clique em **OK**.
O painel de propriedades do usuário é exibido, mostrando o grupo incluído.
f) Selecionar o grupo.
9. Clique em **OK**.
O painel Gerenciamento de Computadores é exibido.

Exibindo quem está em um grupo no Windows
Exiba os membros de um grupo usando o painel de controle.

Procedimento
1. Abra o painel de controle
2. Dê um clique duplo em **Ferramentas Administrativas**.
O painel Ferramentas Administrativas é aberto.
3. Dê um clique duplo em **Gerenciamento de Computadores**.
O painel Gerenciamento de Computadores é aberto.
4. No painel Gerenciamento de Computadores, expanda **Usuários e Grupos Locais**.
5. Selecione **Grupos**.
6. Clique duas vezes em um grupo. O painel de propriedades do grupo é exibido.
O painel de propriedades do grupo é exibido.

Resultados
Os membros do grupo são exibidos.

Removendo um usuário de um grupo no Windows
Remova um usuário de um grupo usando o painel de controle.

Procedimento
1. Abra o painel de controle
2. Dê um clique duplo em **Ferramentas Administrativas**.
O painel Ferramentas Administrativas é aberto.
3. Dê um clique duplo em **Gerenciamento de Computadores**.
O painel Gerenciamento de Computadores é aberto.
4. No painel Gerenciamento de Computadores, expanda **Usuários e Grupos Locais**.
5. Selecione **Usuários**.
6. Clique duas vezes no usuário que você deseja incluir em um grupo.
O painel de propriedades do usuário é exibido.
7. Selecione a guia **Membro de**.
8. Selecione o grupo do qual você deseja remover o usuário e, em seguida, clique em **Remover**.
9. Clique em **OK**.
O painel Gerenciamento de Computadores é exibido.

Resultados
Agora você removeu o usuário do grupo.

Criando e Gerenciando Grupos no HP-UX

No HP-UX, desde que você não esteja usando NIS ou NIS +, use o System Administration Manager (SAM) para trabalhar com grupos.

Criando um grupo no HP-UX

Inclua um usuário em um grupo usando o System Administration Manager

Procedimento

1. No System Administration Manager (SAM), clique duas vezes em Contas para Usuários e Grupos.
2. Clique duas vezes em Grupos.
3. Selecione Incluir no menu suspenso Ações para exibir o painel Incluir um Novo Grupo.
4. Insira o nome do grupo e selecione os usuários que você deseja incluir no grupo.
5. Clique em Aplicar para criar o grupo.

Resultados

Agora você criou um grupo.

Incluindo um usuário em um grupo no HP-UX

Inclua um usuário em um grupo usando o System Administration Manager.

Procedimento

1. No System Administration Manager (SAM), clique duas vezes em Contas para Usuários e Grupos.
2. Clique duas vezes em Grupos.
3. Destaque o nome do grupo e selecione Modificar no menu suspenso Ações para exibir o painel Modificar um Grupo Existente.
4. Selecione um usuário que você deseja incluir no grupo e clique em Incluir.
5. Se desejar incluir outros usuários no grupo, repita a etapa 4 para cada usuário.
6. Quando tiver concluído a inclusão de nomes na lista, clique em OK.

Resultados

Agora você incluiu um usuário em um grupo.

Exibindo quem está em um grupo no HP-UX .

Exiba quem está em um grupo usando o System Administration Manager

Procedimento

1. No System Administration Manager (SAM), clique duas vezes em Contas para Usuários e Grupos.
2. Clique duas vezes em Grupos.
3. Destaque o nome do grupo e selecione Modificar no menu suspenso Ações para exibir o painel Modificar um Grupo Existente, mostrando uma lista dos usuários no grupo.

Resultados

Os membros do grupo são exibidos.

Removendo um Usuário de um Grupo no HP-UX

Remova um usuário de um grupo usando o System Administration Manager.

Procedimento

1. No System Administration Manager (SAM), clique duas vezes em Contas para Usuários e Grupos.
2. Clique duas vezes em Grupos.
3. Destaque o nome do grupo e selecione Modificar no menu suspenso Ações para exibir o painel Modificar um Grupo Existente.
4. Selecione um usuário que você deseja remover do grupo e clique em Remover.

5. Para remover outros usuários do grupo, repita a etapa 4 para cada usuário.
6. Quando tiver concluído a remoção de nomes da lista, clique em OK.

Resultados
Agora você removeu um usuário de um grupo

Criando e Gerenciando Grupos no AIX

No AIX, desde que não esteja usando NIS ou NIS +, use SMITTY para trabalhar com grupos.

Como criar um grupo
Crie um grupo usando SMITTY.

Procedimento

1. No SMITTY, selecione Security and Users e pressione Enter.
2. Selecione Groups e pressione Enter.
3. Selecione Add a Group e pressione Enter.
4. Insira o nome do grupo e os nomes de usuários que você deseja incluir no grupo, separados por vírgulas.
5. Pressione Enter para criar o grupo.

Resultados
Agora você criou um grupo.

Adicionando um Usuário a um Grupo
Inclua um usuário em um grupo usando SMITTY.

Procedimento

1. No SMITTY, selecione Security and Users e pressione Enter.
2. Selecione Groups e pressione Enter.
3. Selecione Change / Show Characteristics of Groups e pressione Enter.
4. Insira o nome do grupo para mostrar uma lista dos membros do grupo.
5. Inclua os nomes dos usuários que você deseja incluir no grupo, separados por vírgulas.
6. Pressione Enter para incluir os nomes no grupo.

Exibindo quem Está em um Grupo
Exiba quem está em um grupo usando SMITTY.

Procedimento

1. No SMITTY, selecione Security and Users e pressione Enter.
2. Selecione Groups e pressione Enter.
3. Selecione Change / Show Characteristics of Groups e pressione Enter.
4. Insira o nome do grupo para mostrar uma lista dos membros do grupo.

Resultados
Os membros do grupo são exibidos.

Removendo um Usuário de um Grupo
Remova um usuário de um grupo usando SMITTY.

Procedimento

1. No SMITTY, selecione Security and Users e pressione Enter.
2. Selecione Groups e pressione Enter.
3. Selecione Change / Show Characteristics of Groups e pressione Enter.
4. Insira o nome do grupo para mostrar uma lista dos membros do grupo.
5. Exclua os nomes dos usuários que você deseja remover do grupo.
6. Pressione Enter para remover os nomes do grupo.

Resultados
Agora você removeu um usuário de um grupo.

Criando e Gerenciando Grupos no Solaris

No Solaris, desde que você não esteja usando NIS ou NIS +, use o arquivo `/etc/group` para trabalhar com grupos.

Criando um grupo no Solaris
Criando um grupo usando o comando **groupadd**.

Procedimento

Digite o seguinte comando: `groupadd` *`group-name`*
em que *group-name* é o nome do grupo.

Resultados
O arquivo `/etc/group` retém informações sobre o grupo.

Incluindo um usuário em um grupo no Solaris
Inclua um usuário em um grupo usando o comando **usermod**.

Procedimento

Para incluir um membro em um grupo complementar, execute o comando `usermod` e liste os grupos complementares dos quais o usuário é um membro atualmente, e os grupos complementares dos quais o usuário deverá se tornar um membro.
Por exemplo, se o usuário for um membro do grupo `groupa` e também se tornar um membro de `groupb`, use o seguinte comando: `usermod -G groupa,groupb` *`user-name`* , em que *user-name* é o nome de usuário.

Exibindo quem está em um grupo no Solaris
Para descobrir quem é um membro de um grupo, examine a entrada desse grupo no arquivo `/etc/group`.

Como remover um usuário de um grupo no Solaris
Remova um usuário de um grupo usando o comando **usermod**.

Procedimento

Para remover um membro de um grupo complementar, execute o comando **usermod** listando os grupos complementares dos quais você deseja que o usuário permaneça um membro.
Por exemplo, se o grupo primário do usuário for `users` e o usuário também for membro dos grupos `mqm`, `groupa` e `groupb`, para remover o usuário do grupo `mqm` , o comando a seguir será usado: `usermod -G groupa,groupb` *`user-name`*, em que *user-name* é o nome do usuário.

Criando e gerenciando grupos no Linux

No Linux, desde que você não esteja usando NIS ou NIS+, use o arquivo `/etc/group` para trabalhar com grupos.

Criando um grupo no Linux .

Crie um grupo usando o comando **groupadd**.

Procedimento

Para criar um novo grupo, digite o comando a seguir: `groupadd -g group-ID group-name`

, em que *group-ID* é o identificador numérico do grupo e *group-name* é o nome do grupo.

Resultados

O arquivo `/etc/group` retém informações sobre o grupo.

Incluindo um usuário em um grupo no Linux

Inclua um usuário em um grupo usando o comando **usermod**.

Procedimento

Para incluir um membro em um grupo complementar, execute o comando `usermod` e liste os grupos complementares dos quais o usuário é um membro atualmente, e os grupos complementares dos quais o usuário deverá se tornar um membro.
Por exemplo, se o usuário for um membro do grupo `groupa` e também se tornar um membro de `groupb`, o comando a seguir será usado: `usermod -G groupa,groupb user-name`

, em que *user-name* é o nome de usuário.

Exibindo quem está em um grupo em Linux

Exiba quem está em um grupo usando o comando **getent**.

Procedimento

Para exibir quem é membro de um grupo, digite o comando a seguir: `getent group group-name`

, em que *group-name* é o nome do grupo.

Removendo um Usuário de um Grupo

Remova um usuário de um grupo usando o comando **usermod**.

Procedimento

Para remover um membro de um grupo complementar, execute o comando **usermod** listando os grupos complementares dos quais você deseja que o usuário permaneça um membro.
Por exemplo, se o grupo primário do usuário for `users` e o usuário também for um membro dos grupos `mqm`, `groupa` e `groupb`, para remover o usuário do grupo `mqm`, o comando a seguir será usado: `usermod -G groupa,groupb user-name`

, em que *user-name* é o nome de usuário.

Como as Autorizações Funcionam

As tabelas de especificação de autorização nos tópicos desta seção definem precisamente como as autorizações funcionam e as restrições que se aplicam.

As tabelas aplicam-se a estas situações:

- Aplicativos que emitem chamadas MQI
- Programas de administração que emitem comandos MQSC como PCFs Escape

- Programas de administração que emitem comando de PCF

Nesta seção, as informações são apresentadas como um conjunto de tabelas que especificam o seguinte:

Ação a ser executada
Opção de MQI, comando MQSC ou comando PCF.

Objeto de controle de acesso
Fila, processo, gerenciador de filas, lista de nomes, informações sobre autenticação, canal, canal de conexão do cliente, listener ou serviço.

Autorização necessária
Expressa como uma constante MQZAO_.

Nas tabelas, as constantes prefixadas com MQZAO_ correspondem às palavras-chave na lista de autorização para o comando `setmqaut` da entidade específica. Por exemplo, MQZAO_BROWSE corresponde à palavra-chave `+browse`, MQZAO_SET_ALL_CONTEXT corresponde à palavra-chave `+setall`, etc. Essas constantes são definidas no arquivo de cabeçalho cmqzc.h, fornecido com o produto.

Autorizações para Chamadas MQI

MQCONN, **MQOPEN**, **MQPUT1** e **MQCLOSE** podem requerer verificações de autorização. As tabelas deste tópico resumem as autorizações necessárias para cada chamada.

Um aplicativo terá permissão para emitir chamadas MQI e opções específicas somente se o identificador de usuário sob o qual estiver sendo executado (ou cujas autorizações puder assumir) tiver recebido a autorização relevante.

Quatro chamadas MQI podem requerer verificações de autorização: **MQCONN**, **MQOPEN**, **MQPUT1** e **MQCLOSE**.

Para MQOPEN e MQPUT1, a verificação de autoridade é feita no nome do objeto que está sendo aberto e não no nome ou nomes resultantes após a resolução de um nome. Por exemplo, um aplicativo pode receber autoridade para abrir uma fila de alias sem ter autoridade para abrir a fila de base para a qual o alias é resolvido. A regra é que a verificação seja realizada na primeira definição encontrada durante o processo de resolução de um nome que não é um alias do gerenciador de filas, a menos que a definição de alias do gerenciador de filas seja aberta diretamente; ou seja, seu nome é exibido no campo *ObjectName* do descritor de objeto. A autoridade é sempre necessária para o objeto que está sendo aberto. Em alguns casos, autoridade adicional independente da fila, obtida por meio de uma autorização para o objeto de gerenciador de filas, é necessária.

Tabela 8 na página 91, Tabela 9 na página 92, Tabela 10 na página 92 e Tabela 11 na página 93 resumem as autorizações necessárias para cada chamada. Nas tabelas, *Não aplicável* significa que a verificação de autorização não é relevante para essa operação; *Nenhuma verificação* significa que nenhuma verificação de autorização é executada.

Nota: Você não encontrará nenhuma menção a listas de nomes, canais, canais de conexão do cliente, listeners, serviços ou objetos de informações sobre autenticação nessas tabelas. Isso é porque nenhuma das autorizações se aplica a esses objetos, exceto MQOO_INQUIRE, para o qual as mesmas autorizações se aplicam como para os outros objetos.

A autorização especial MQZAO_ALL_MQI inclui todas as autorizações nas tabelas que são relevantes ao tipo de objeto, exceto MQZAO_DELETE e MQZAO_DISPLAY, que são classificados como autorizações de administração.

Para modificar qualquer uma das opções de contexto da mensagem, deve-se ter as autorizações apropriadas para emitir a chamada. Por exemplo, para usar MQOO_SET_IDENTITY_CONTEXT ou MQPMO_SET_IDENTITY_CONTEXT, deve-se ter a permissão `+setid`.

Tabela 8. Autorização de segurança necessária para chamadas MQCONN

Autorização necessária para:	**Objeto da fila (“1” na página 93).**	**Objeto de processo**	**Objeto do gerenciador de filas**
MQCONN	Não-aplicável	Não-aplicável	MQZAO_CONNECT

Tabela 9. Autorização de segurança necessária para chamadas MQOPEN			
Autorização necessária para:	**Objeto da fila (“1” na página 93).**	**Objeto de processo**	**Objeto do gerenciador de filas**
MQOO_INQUIRE	MQZAO_INQUIRE	MQZAO_INQUIRE	MQZAO_INQUIRE
MQOO_BROWSE	MQZAO_BROWSE	Não-aplicável	Sem verificação
MQOO_INPUT_*	MQZAO_INPUT	Não-aplicável	Sem verificação
MQOO_SAVE_ ALL_CONTEXT **(“2” na página 93)**	MQZAO_INPUT	Não-aplicável	Não-aplicável
MQOO_OUTPUT (Fila normal) **(“3” na página 93)**	MQZAO_OUTPUT	Não-aplicável	Não-aplicável
MQOO_PASS_ IDENTITY_CONTEXT **(“4” na página 93)**	MQZAO_PASS_ IDENTITY_CONTEXT	Não-aplicável	Sem verificação
MQOO_PASS_ALL_ CONTEXT **(“4” na página 93, “5” na página 93)**	MQZAO_PASS _ALL_CONTEXT	Não-aplicável	Sem verificação
MQOO_SET_ IDENTITY_CONTEXT **(“4” na página 93, “5” na página 93)**	MQZAO_SET_ IDENTITY_CONTEXT	Não-aplicável	MQZAO_SET_ IDENTITY_CONTEXT **(“6” na página 93)**
MQOO_SET_ ALL_CONTEXT **(“4” na página 93, “7” na página 93)** .	MQZAO_SET_ ALL_CONTEXT	Não-aplicável	MQZAO_SET_ ALL_CONTEXT **(“6” na página 93)**
MQOO_OUTPUT (Fila de transmissão) **(“8” na página 93)**	MQZAO_SET_ ALL_CONTEXT	Não-aplicável	MQZAO_SET_ ALL_CONTEXT **(“6” na página 93)**
MQOO_SET	MQZAO_SET	Não-aplicável	Sem verificação
MQOO_ALTERNATE_ USER_AUTHORITY	**(“9” na página 93)**	**(“9” na página 93)**	MQZAO_ALTERNATE_ USER_AUTHORITY **(“9” na página 93, “10” na página 93)**

Tabela 10. Autorização de segurança necessária para chamadas MQPUT1			
Autorização necessária para:	**Objeto da fila (“1” na página 93).**	**Objeto de processo**	**Objeto do gerenciador de filas**
MQPMO_PASS_ IDENTITY_CONTEXT	MQZAO_PASS_ IDENTITY_CONTEXT **(“11” na página 93)**	Não-aplicável	Sem verificação
MQPMO_PASS_ALL _CONTEXT	MQZAO_PASS_ ALL_CONTEXT **(“11” na página 93)**	Não-aplicável	Sem verificação
MQPMO_SET_ IDENTITY_CONTEXT	MQZAO_SET_ IDENTITY_CONTEXT **(“11” na página 93)**	Não-aplicável	MQZAO_SET_ IDENTITY_CONTEXT **(“6” na página 93)**

Tabela 10. Autorização de segurança necessária para chamadas MQPUT1 (continuação)			
Autorização necessária para:	**Objeto da fila (“1” na página 93).**	**Objeto de processo**	**Objeto do gerenciador de filas**
MQPMO_SET_ ALL_CONTEXT	MQZAO_SET_ ALL_CONTEXT **(“11” na página 93)**	Não-aplicável	MQZAO_SET_ ALL_CONTEXT **(“6” na página 93)**
(Fila de transmissão) **(“8” na página 93)**	MQZAO_SET_ ALL_CONTEXT	Não-aplicável	MQZAO_SET_ ALL_CONTEXT **(“6” na página 93)**
MQPMO_ALTERNATE_ USER_AUTHORITY	**(“12” na página 94)**	Não-aplicável	MQZAO_ALTERNATE_ USER_AUTHORITY **(“10” na página 93)**

Tabela 11. Autorização de segurança necessária para chamadas MQCLOSE			
Autorização necessária para:	**Objeto da fila (“1” na página 93).**	**Objeto de processo**	**Objeto do gerenciador de filas**
MQCO_DELETE	MQZAO_DELETE **(“13” na página 94)**	Não-aplicável	Não-aplicável
MQCO_DELETE _PURGE	MQZAO_DELETE **(“13” na página 94)**	Não-aplicável	Não-aplicável

Notas para as tabelas:

1. Se for abrir uma fila modelo:
   - A autoridade MQZAO_DISPLAY será necessária para a fila modelo, além da autoridade para abrir a fila modelo para o tipo de acesso para o qual você está abrindo.
   - A autoridade MQZAO_CREATE não é necessária para criar o fila dinâmica.
   - O identificador de usuários usado para abrir a fila modelo recebe automaticamente todas as autoridades específicas da fila (equivalente a MQZAO_ALL) para a fila dinâmica criada.
2. MQOO_INPUT_* também deve ser especificado. Isso é válido para uma fila local, modelo ou de alias.
3. Essa verificação é executada para todos os casos de saída, exceto para filas de transmissão (consulte a nota “8” na página 93).
4. MQOO_OUTPUT também deve ser especificado.
5. MQOO_PASS_IDENTITY_CONTEXT também é sugerido por essa opção.
6. Essa autoridade é necessária para o objeto de gerenciador de filas e a fila específica.
7. MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT e MQOO_SET_IDENTITY_CONTEXT também são sugeridos por essa opção.
8. Essa verificação é executada para uma fila local ou modelo que tem um atributo de fila *Usage* de MQUS_TRANSMISSION e está sendo aberta diretamente para saída. Ela não será aplicada se uma fila remota estiver sendo aberta (especificando-se os nomes do gerenciador de filas remotas e da fila remota ou especificando-se o nome de uma definição local da fila remota).
9. Pelo menos um de MQOO_INQUIRE (para qualquer tipo de objeto) ou MQOO_BROWSE, MQOO_INPUT_*, MQOO_OUTPUT ou MQOO_SET (para filas) também deve ser especificado. A verificação executada é como para as outras opções especificadas, usando-se o identificador de usuário alternativo fornecido para a autoridade de objeto com nome específico, e a autoridade de aplicativo atual para a verificação MQZAO_ALTERNATE_USER_IDENTIFIER.
10. Essa autorização permite que qualquer *AlternateUserId* seja especificado.
11. Uma verificação MQZAO_OUTPUT também será executada se a fila não tiver um atributo de fila *Usage* de MQUS_TRANSMISSION.

12. A verificação executada é como para as outras opções especificadas, usando-se o identificador de usuário alternativo fornecido para a autoridade de fila com nome específico, e a autoridade de aplicativo atual para a verificação MQZAO_ALTERNATE_USER_IDENTIFIER.
13. A verificação será executada apenas se ambos os itens a seguir forem verdadeiros:
    - Uma fila dinâmica permanente está sendo fechada e excluída.
    - A fila não foi criada pela chamada MQOPEN que retornou a manipulação de objetos que estava sendo usada.

    Caso contrário, não haverá verificação.

Autorizações para Comandos MQSC em PCFs Escape

Estas informações resumem as autorizações necessárias para cada comando MQSC contido no PCF Escape.

Não aplicável significa que esta operação não é relevante para esse tipo de objeto.

O ID de usuário com o qual o programa que envia o comando está sendo executado também deve ter as seguintes autoridades:

- Autoridade MQZAO_CONNECT para o gerenciador de filas
- Autoridade MQZAO_DISPLAY no gerenciador de filas para executar comandos de PCF
- Autoridade para emitir o comando MQSC dentro do texto do comando PCF Escape

ALTER *object*

Object	**Autorização necessária**
Fila	MQZAO_CHANGE
Tópico	MQZAO_CHANGE
Processo	MQZAO_CHANGE
Gerenciador de Filas	MQZAO_CHANGE
Lista de Nomes	MQZAO_CHANGE
Informações sobre Autenticação	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexão do cliente	MQZAO_CHANGE
Listener	MQZAO_CHANGE
Serviço	MQZAO_CHANGE
Informações de comunicação	MQZAO_CHANGE

CLEAR *object*

Object	**Autorização necessária**
Fila	MQZAO_CLEAR
Tópico	MQZAO_CLEAR
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	Não-aplicável

Object	Autorização necessária
Canal de conexão do cliente	Não-aplicável
Listener	Não-aplicável
Serviço	Não-aplicável
Informações de comunicação	Não-aplicável

DEFINE *object* NOREPLACE (“1” na página 98)

Object	Autorização necessária
Fila	MQZAO_CREATE **(“2” na página 98)**
Tópico	MQZAO_CREATE **(“2” na página 98)**
Processo	MQZAO_CREATE **(“2” na página 98)**
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CREATE **(“2” na página 98)**
Informações sobre Autenticação	MQZAO_CREATE **(“2” na página 98)**
Canal	MQZAO_CREATE **(“2” na página 98)**
Canal de conexão do cliente	MQZAO_CREATE **(“2” na página 98)**
Listener	MQZAO_CREATE **(“2” na página 98)**
Serviço	MQZAO_CREATE **(“2” na página 98)**
Informações de comunicação	MQZAO_CREATE **(“2” na página 98)**

DEFINE *object* REPLACE (“1” na página 98, “3” na página 99)

Object	Autorização necessária
Fila	MQZAO_CHANGE
Tópico	MQZAO_CHANGE
Processo	MQZAO_CHANGE
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CHANGE
Informações sobre Autenticação	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexão do cliente	MQZAO_CHANGE
Listener	MQZAO_CHANGE
Serviço	MQZAO_CHANGE
Informações de comunicação	MQZAO_CHANGE

EXCLUIR *object*

Object	Autorização necessária
Fila	MQZAO_DELETE
Tópico	MQZAO_DELETE

Object	Autorização necessária
Processo	MQZAO_DELETE
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_DELETE
Informações sobre Autenticação	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexão do cliente	MQZAO_DELETE
Listener	MQZAO_DELETE
Serviço	MQZAO_DELETE
Informações de comunicação	MQZAO_DELETE

DISPLAY *object*

Object	Autorização necessária
Fila	MQZAO_DISPLAY
Tópico	MQZAO_DISPLAY
Processo	MQZAO_DISPLAY
Gerenciador de Filas	MQZAO_DISPLAY
Lista de Nomes	MQZAO_DISPLAY
Informações sobre Autenticação	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexão do cliente	MQZAO_DISPLAY
Listener	MQZAO_DISPLAY
Serviço	MQZAO_DISPLAY
Informações de comunicação	MQZAO_DISPLAY

START *object*

Object	Autorização necessária
Fila	Não-aplicável
Tópico	Não-aplicável
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	MQZAO_CONTROL
Canal de conexão do cliente	Não-aplicável
Listener	MQZAO_CONTROL
Serviço	MQZAO_CONTROL

Object	Autorização necessária
Informações de comunicação	Não-aplicável

PARAR *object*

Object	Autorização necessária
Fila	Não-aplicável
Tópico	Não-aplicável
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	MQZAO_CONTROL
Canal de conexão do cliente	Não-aplicável
Listener	MQZAO_CONTROL
Serviço	MQZAO_CONTROL
Informações de comunicação	Não-aplicável

Comandos do Canal

Comando:	Object	Autorização necessária
PING CHANNEL	Canal	MQZAO_CONTROL
RESET CHANNEL	Canal	MQZAO_CONTROL_EXTENDED
RESOLVE CHANNEL	Canal	MQZAO_CONTROL_EXTENDED

Comandos de Assinatura

Comando:	Object	Autorização necessária
ALTER SUB	Tópico	MQZAO_CONTROL
DEFINE SUB	Tópico	MQZAO_CONTROL
DELETE SUB	Tópico	MQZAO_CONTROL
DISPLAY SUB	Tópico	MQZAO_DISPLAY

Comandos de Segurança

Comando:	Object	Autorização necessária
SET AUTHREC	Gerenciador de Filas	MQZAO_CHANGE
DELETE AUTHREC	Gerenciador de Filas	MQZAO_CHANGE
DISPLAY AUTHREC	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY AUTHSERV	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY ENTAUTH	Gerenciador de Filas	MQZAO_DISPLAY
SET CHLAUTH	Gerenciador de Filas	MQZAO_CHANGE

Comando:	Object	Autorização necessária
DISPLAY CHLAUTH	Gerenciador de Filas	MQZAO_DISPLAY
REFRESH SECURITY	Gerenciador de Filas	MQZAO_CHANGE

Exibições de status

Comando:	Object	Autorização necessária
DISPLAY CHSTATUS	Gerenciador de Filas	MQZAO_DISPLAY Observe que a autoridade `+inq` (ou equivalentemente MQZAO_INQUIRE) será necessária na fila de transmissão se o tipo de canal for CLUSSDR.
DISPLAY LSSTATUS	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY PUBSUB	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY SBSTATUS	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY SVSTATUS	Gerenciador de Filas	MQZAO_DISPLAY
DISPLAY TPSTATUS	Gerenciador de Filas	MQZAO_DISPLAY

Comandos do Cluster

Comando:	Object	Autorização necessária
EXIBIR CLUSQMGR	Gerenciador de Filas	MQZAO_DISPLAY
REFRESH CLUSTER	associação ao grupo 'mqm' necessária	
RESET CLUSTER	associação ao grupo 'mqm' necessária	
SUSPEND QMGR	associação ao grupo 'mqm' necessária	
RESUME QMGR	associação ao grupo 'mqm' necessária	

Outros comandos administrativos

Comando:	Object	Autorização necessária
PING QMGR	Gerenciador de Filas	MQZAO_DISPLAY
REFRESH QMGR	Gerenciador de Filas	MQZAO_CHANGE
RESET QMGR	Gerenciador de Filas	MQZAO_CHANGE
DISPLAY CONN	Gerenciador de Filas	MQZAO_DISPLAY
STOP CONN	Gerenciador de Filas	MQZAO_CHANGE

Nota:

1. Para comandos DEFINE, a autoridade MQZAO_DISPLAY também será necessária para o objeto LIKE, se houver um especificado, ou no objeto SYSTEM.DEFAULT.xxx apropriado, se LIKE for omitido.
2. A autoridade MQZAO_CREATE não é específica a um determinado objeto ou tipo de objeto. A autoridade Criar é concedida para todos os objetos de um gerenciador de filas especificado, especificando-se um tipo de objeto de QMGR no comando `setmqaut`.

3. Isso se aplicará se o objeto a ser substituído já existir. Caso contrário, a verificação será como para DEFINE *object* NOREPLACE.

Informações relacionadas

Armazenamento em Cluster: Usando Melhores Práticas de REFRESH CLUSTER

Autorizações para Comandos PCF

Esta seção resume as autorizações necessárias para cada comando PCF.

Nenhuma verificação significa que nenhuma verificação de autorização é executada; *Não aplicável* significa que esta operação não é relevante para este tipo de objeto.

O ID de usuário com o qual o programa que envia o comando está sendo executado também deve ter as seguintes autoridades:

- Autoridade MQZAO_CONNECT para o gerenciador de filas
- Autoridade MQZAO_DISPLAY no gerenciador de filas para executar comandos de PCF

A autorização especial MQZAO_ALL_ADMIN inclui todas as autorizações na lista a seguir que são relevantes ao tipo de objeto, exceto MQZAO_CREATE, que não é específica a um determinado objeto ou tipo de objeto.

Mudar *object*

Object	**Autorização necessária**
Fila	MQZAO_CHANGE
Tópico	MQZAO_CHANGE
Processar	MQZAO_CHANGE
Gerenciador de Filas	MQZAO_CHANGE
Lista de Nomes	MQZAO_CHANGE
Informações sobre Autenticação	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexão do cliente	MQZAO_CHANGE
Receptor	MQZAO_CHANGE
Serviço	MQZAO_CHANGE
Informações de Comunicação	MQZAO_CHANGE

Clear *object*

Object	**Autorização necessária**
Fila	MQZAO_CLEAR
Tópico	MQZAO_CLEAR
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	Não-aplicável
Canal de conexão do cliente	Não-aplicável

Object	Autorização necessária
Listener	Não-aplicável
Serviço	Não-aplicável
Informações de comunicação	Não-aplicável

Copiar *objeto* (sem substituir) (1)

Object	Autorização necessária
Fila	MQZAO_CREATE **(2)**
Tópico	MQZAO_CREATE **(2)**
Processar	MQZAO_CREATE **(2)**
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CREATE **(2)**
Informações sobre Autenticação	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**
Canal de conexão do cliente	MQZAO_CREATE **(2)**
Receptor	MQZAO_CREATE **(2)**
Serviço	MQZAO_CREATE **(2)**
Informações de Comunicação	MQZAO_CREATE **(“2” na página 105)**

Copie *objeto* (com substituição) (1, 4)

Object	Autorização necessária
Fila	MQZAO_CHANGE
Tópico	MQZAO_CHANGE
Processar	MQZAO_CHANGE
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CHANGE
Informações sobre Autenticação	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexão do cliente	MQZAO_CHANGE
Receptor	MQZAO_CHANGE
Serviço	MQZAO_CHANGE
Informações de Comunicação	MQZAO_CHANGE

Crie *objeto* (sem substituir) (3) .

Object	Autorização necessária
Fila	MQZAO_CREATE **(2)**
Tópico	MQZAO_CREATE **(2)**
Processar	MQZAO_CREATE **(2)**

Object	Autorização necessária
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CREATE **(2)**
Informações sobre Autenticação	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**
Canal de conexão do cliente	MQZAO_CREATE **(2)**
Receptor	MQZAO_CREATE **(2)**
Serviço	MQZAO_CREATE **(2)**
Informações de Comunicação	MQZAO_CREATE **(2)**

Criar *objeto* (com substituição) (3, 4)

Object	Autorização necessária
Fila	MQZAO_CHANGE
Tópico	MQZAO_CHANGE
Processar	MQZAO_CHANGE
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_CHANGE
Informações sobre Autenticação	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexão do cliente	MQZAO_CHANGE
Receptor	MQZAO_CHANGE
Serviço	MQZAO_CHANGE
Informações de Comunicação	MQZAO_CHANGE

Excluir *object*

Object	Autorização necessária
Fila	MQZAO_DELETE
Tópico	MQZAO_DELETE
Processar	MQZAO_DELETE
Gerenciador de Filas	Não-aplicável
Lista de Nomes	MQZAO_DELETE
Informações sobre Autenticação	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexão do cliente	MQZAO_DELETE
Receptor	MQZAO_DELETE
Serviço	MQZAO_DELETE
Informações de Comunicação	MQZAO_DELETE

Inquire *object*

Object	Autorização necessária
Fila	MQZAO_DISPLAY
Tópico	MQZAO_DISPLAY
Processar	MQZAO_DISPLAY
Gerenciador de Filas	MQZAO_DISPLAY
Lista de Nomes	MQZAO_DISPLAY
Informações sobre Autenticação	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexão do cliente	MQZAO_DISPLAY
Receptor	MQZAO_DISPLAY
Serviço	MQZAO_DISPLAY
Informações de Comunicação	MQZAO_DISPLAY

Inquire *object* names

Object	Autorização necessária
Fila	Sem verificação
Tópico	Sem verificação
Processo	Sem verificação
Gerenciador de Filas	Sem verificação
Lista de Nomes	Sem verificação
Informações sobre Autenticação	Sem verificação
Canal	Sem verificação
Canal de conexão do cliente	Sem verificação
Listener	Sem verificação
Serviço	Sem verificação
Informações de comunicação	Sem verificação

Iniciar *object*

Object	Autorização necessária
Fila	Não-aplicável
Tópico	Não-aplicável
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	MQZAO_CONTROL

Object	Autorização necessária
Canal de conexão do cliente	Não-aplicável
Receptor	MQZAO_CONTROL
Serviço	MQZAO_CONTROL
Informações de comunicação	Não-aplicável

Parar *object*

Object	Autorização necessária
Fila	Não-aplicável
Tópico	Não-aplicável
Processo	Não-aplicável
Gerenciador de Filas	Não-aplicável
Lista de Nomes	Não-aplicável
Informações sobre Autenticação	Não-aplicável
Canal	MQZAO_CONTROL
Canal de conexão do cliente	Não-aplicável
Receptor	MQZAO_CONTROL
Serviço	MQZAO_CONTROL
Informações de comunicação	Não-aplicável

Comandos do Canal

Comando:	Object	Autorização necessária
Executar ping no Canal	Canal	MQZAO_CONTROL
Redefinir Canal	Canal	MQZAO_CONTROL_EXTENDED
Resolver Canal	Canal	MQZAO_CONTROL_EXTENDED

Comandos de Assinatura

Comando:	Object	Autorização necessária
Change Subscription	Tópico	MQZAO_CONTROL
Criar assinatura	Tópico	MQZAO_CONTROL
Excluir assinatura	Tópico	MQZAO_CONTROL
Consultar Assinatura	Tópico	MQZAO_DISPLAY

Comandos de Segurança

Comando:	Object	Autorização necessária
Configurar Registro de Autoridade	Gerenciador de Filas	MQZAO_CHANGE
Excluir Registro de Autoridade	Gerenciador de Filas	MQZAO_CHANGE

Comando:	Object	Autorização necessária
Consultar Registros de Autoridade	Gerenciador de Filas	MQZAO_DISPLAY
Consultar Autoridade de Serviço	Gerenciador de Filas	MQZAO_DISPLAY
Solicitar Autoridade de Entidade	Gerenciador de Filas	MQZAO_DISPLAY
Configurar Registro de Autenticação de Canal	Gerenciador de Filas	MQZAO_CHANGE
Solicitar Registros de Autenticação de Canal	Gerenciador de Filas	MQZAO_DISPLAY
Atualizar segurança	Gerenciador de Filas	MQZAO_CHANGE

Exibições de status

Comando:	Object	Autorização necessária
Consultar Status do Canal	Gerenciador de Filas	MQZAO_DISPLAY Observe que a autoridade `+inq` (ou equivalentemente MQZAO_INQUIRE) será necessária na fila de transmissão se o tipo de canal for CLUSSDR.
Consulte o status do ouvinte de canal	Gerenciador de Filas	MQZAO_DISPLAY
Investigar Status da Pub/Ass	Gerenciador de Filas	MQZAO_DISPLAY
Inquire Subscription Status	Gerenciador de Filas	MQZAO_DISPLAY
Consultar Status do Serviço	Gerenciador de Filas	MQZAO_DISPLAY
Inquire Topic Status	Gerenciador de Filas	MQZAO_DISPLAY

Comandos do Cluster

Comando:	Object	Autorização necessária
Consultar Gerenciador de Filas de Clusters	Gerenciador de Filas	MQZAO_DISPLAY
Refresh Cluster	associação ao grupo 'mqm' necessária	
Reset Cluster	associação ao grupo 'mqm' necessária	
Suspender Cluster de Gerenciador de Filas	associação ao grupo 'mqm' necessária	
Retomar Cluster de Gerenciador de Filas	associação ao grupo 'mqm' necessária	

Outros comandos administrativos

Comando:	Object	Autorização necessária
Executar Ping do Gerenciador de Filas	Gerenciador de Filas	MQZAO_DISPLAY
Atualizar Gerenciador de Filas	Gerenciador de Filas	MQZAO_CHANGE

Comando:	Object	Autorização necessária
Reconfigurar Gerenciador de Filas	Gerenciador de Filas	MQZAO_CHANGE
Reconfigurar as Estatísticas de Fila	Fila	MQZAO_DISPLAY e MQZAO_CHANGE
Consultar Conexão	Gerenciador de Filas	MQZAO_DISPLAY
Para Conexão	Gerenciador de Filas	MQZAO_CHANGE

Nota:

1. Para comandos Copy, a autoridade MQZAO_DISPLAY também é necessária para o objeto De.
2. A autoridade MQZAO_CREATE não é específica a um determinado objeto ou tipo de objeto. A autoridade Criar é concedida para todos os objetos de um gerenciador de filas especificado, especificando-se um tipo de objeto de QMGR no comando `setmqaut`.
3. Para comandos de Criação, a autoridade MQZAO_DISPLAY também é necessária para o SYSTEM.DEFAULT.* objeto.
4. Isso se aplicará se o objeto a ser substituído já existir. Caso contrário, a verificação será como para Copy ou Create sem substituição.

Considerações especiais para segurança no Windows

Algumas funções de segurança se comportam de forma diferente em diferentes versões do Windows..

A segurança do IBM WebSphere MQ depende de chamadas para a API do sistema operacional para obter informações sobre autorizações de usuários e associações de grupo. Algumas funções não se comportam identicamente nos sistemas Windows . Esta coleção de tópicos inclui descrições de como essas diferenças podem afetar a IBM WebSphere MQ segurança quando você estiver executando IBM WebSphere MQ em um ambiente do Windows .

O Programa de Saída de Canal da SSPI

O WebSphere MQ para Windows fornece um programa de entrada de segurança, que pode ser usado nos canais de mensagens e MQI. A saída é fornecida como código-fonte e de objeto, além de fornecer autenticação unilateral e de duas vias.

A saída de segurança usa a Security Support Provider Interface (SSPI), que fornece os recursos de segurança integrados de plataformas Windows ..

A saída de segurança fornece os seguintes serviços de identificação e de autenticação:

Autenticação de uma via

Isso usa o suporte de autenticação do Windows NT LAN Manager (NTLM). O NTLM permite que os servidores autentiquem seus clientes. Ele não permite que um cliente autentique um servidor, ou um servidor autentique outro servidor. O NTLM foi projetado para um ambiente de rede no qual supõe-se que os servidores sejam autênticos. NTLM é suportado em todas as plataformas Windows que são suportadas pelo WebSphere MQ Versão 7.0.

Esse serviço geralmente é usado em um canal MQI para permitir que um gerenciador de filas do servidor autentique um aplicativo cliente MQI do WebSphere MQ . Um aplicativo cliente é identificado pelo ID de usuário associado ao processo em execução.

Para efetuar a autenticação, a saída de segurança na extremidade do cliente de um canal adquire um token de autenticação do NTLM e envia o token em uma mensagem de segurança para seu parceiro na outra extremidade do canal. A saída de segurança do parceiro transmite o token para o NTLM, que verifica se ele é autêntico. Se a saída de segurança do parceiro não for atendida quanto à autenticidade do token, ela instrui o MCA a fechar o canal.

Autenticação de duas vias ou mútua

Utiliza os serviços de autenticação do Kerberos. O protocolo Kerberos não supõe que os servidores em um ambiente de rede sejam autênticos. Servidores podem autenticar clientes e outros servidores, e clientes podem autenticar servidores. Kerberos é suportado em todas as plataformas Windows suportadas pelo WebSphere MQ Versão 7.0.

Este serviço pode ser utilizado em canais de mensagens e do MQI. Em um canal de mensagens, ele fornece autenticação mútua dos dois gerenciadores de filas. Em um canal MQI, ele permite que o gerenciador de filas do servidor e o aplicativo cliente MQI do WebSphere MQ se autentiquem. Um gerenciador de filas é identificado por seu nome prefixado pela sequência `ibmMQSeries/`. Um aplicativo cliente é identificado pelo ID de usuário associado ao processo em execução.

Para efetuar a autenticação mútua, a saída de segurança iniciadora adquire um token de autenticação do servidor de segurança Kerberos e envia o token em uma mensagem de segurança para seu parceiro. A saída de segurança do parceiro transmite o token para o servidor de segurança do Kerberos, que verifica se é autêntico. O servidor de segurança do Kerberos gera um segundo token, que é enviado pelo parceiro em uma mensagem de segurança para a saída de segurança iniciadora. A saída de segurança iniciadora solicita então ao servidor Kerberos que verifique se o segundo token é autêntico. Durante esta troca, se alguma das saídas de segurança não for atendida quanto à autenticidade do token enviado pela outra, ela instrui o MCA a fechar o canal.

A saída de segurança é fornecida no formato de fonte e de objeto. Você pode utilizar o código fonte como um ponto de início para criar seus próprios programas de saída de canal ou utilizar o módulo de objeto conforme fornecido. O módulo de objeto tem dois pontos de entrada, uma para autenticação de uma via utilizando o suporte para autenticação do NTLM e o outro para autenticação de duas vias utilizando os serviços de autenticação do Kerberos.

Para obter mais informações sobre como o programa de saída do canal SSPI funciona e para obter instruções sobre como implementá-lo, consulte Usando a saída de segurança SSPI em sistemas Windows.

Quando você obtém um erro 'grupo não localizado' no Windows

Esse problema pode surgir porque o WebSphere MQ perde o acesso ao grupo mqm local quando servidores Windows são promovidos ou rebaixados de controladores de domínio. Para resolver esse problema, recrie o grupo mqm local.

O sintoma é um erro indicando a perda de um grupo mqm local, por exemplo:

```
>crtmqm qm0
AMQ8066:Local mqm group not found.
```

Alterar o estado de uma máquina entre o servidor e o controlador de domínio pode afetar a operação do WebSphere MQ, porque o WebSphere MQ usa um grupo mqm definido localmente. Quando um servidor é promovido para controlador de domínio, o escopo é alterado de local para domínio local. Quando a máquina é rebaixada para servidor, todos os grupos locais do domínio são removidos. Isso significa que alterar uma máquina de servidor para controlador de domínio e novamente para servidor perde o acesso a um grupo mqm local.

Para remediar esse problema, recrie o grupo mqm local usando as ferramentas de gerenciamento padrão do Windows . Como todas as informações de associação ao grupo são perdidas, deve-se restabelecer usuários privilegiados do WebSphere MQ no grupo mqm local recém-criado. Se a máquina for um membro de domínio, você também deverá incluir o grupo mqm de domínio no grupo mqm local para conceder ao domínio privilegiado WebSphere MQ os IDs de usuário do nível de autoridade necessário

Quando você tiver problemas com IBM WebSphere MQ e controladores de domínio no Windows

Certos problemas podem surgir com as configurações de segurança quando os servidores Windows são promovidos para controladores de domínio

Ao promover servidores Windows 2000, Windows 2003 ou Windows Server 2008 para controladores de domínio, é apresentada a opção de selecionar uma configuração de segurança padrão ou não padrão

relacionada a permissões de usuário e grupo. Essa opção controla se usuários arbitrários poderão recuperar associações do grupo do Active Directory. Como o WebSphere MQ depende de informações de associação ao grupo para implementar sua política de segurança, é importante que o ID do usuário que está executando operações do WebSphere MQ possa determinar as associações ao grupo de outros usuários.

No Windows 2000, quando um domínio é criado usando a opção de segurança padrão, o ID do usuário padrão criado pelo WebSphere MQ durante o processo de instalação pode obter associações ao grupo para outros usuários conforme necessário. O produto, então, é instalado normalmente, criando objetos padrão, e o gerenciador de filas pode determinar a autoridade de acesso de usuários locais e do domínio, se necessário.

No Janelas 2000, quando um domínio é criado usando a opção de segurança não padrão ou no Janelas 2003 e Janelas Server 2008 quando um domínio é criado usando a opção de segurança padrão, o ID do usuário criado pelo WebSphere MQ durante a instalação nem sempre pode determinar as associações ao grupo necessárias. Neste caso, você precisa saber:

- Como o Windows 2000 com permissões de segurança não padrão ou Windows 2003 e Windows Server 2008 com padrão se comporta
- Como permitir que membros do grupo mqm do domínio leiam a associação ao grupo
- Como configurar um serviço IBM WebSphere MQ Windows para executar em um usuário do domínio

Domínio do Windows 2000 com não padrão ou domínio do Windows 2003 e do Windows Server 2008 com permissões de segurança padrão
A instalação do WebSphere MQ se comporta de forma diferente nesses sistemas operacionais, dependendo de um usuário local ou um usuário do domínio executar a instalação.

Se um usuário **local** instalar o WebSphere MQ, o Assistente Preparar WebSphere MQ detectará que o usuário local criado para o serviço IBM WebSphere MQ Windows pode recuperar as informações de associação ao grupo do usuário de instalação. O assistente Preparar WebSphere MQ faz perguntas ao usuário sobre a configuração de rede para determinar se há outras contas do usuário definidas em controladores de domínio em execução no Windows 2000 ou mais recente Se sim, o serviço do IBM WebSphere MQ Windows precisar ser executado em uma conta de usuário do domínio com configurações e autoridades específicas. O Assistente Preparar WebSphere MQ solicita ao usuário os detalhes da conta desse usuário. Sua ajuda online fornece detalhes da conta de usuário de domínio necessária que podem ser enviados ao administrador de domínio.

Se um usuário do **domínio** instalar WebSphere MQ, o Assistente Preparar WebSphere MQ detectará que o usuário local criado para o serviço IBM WebSphere MQ Windows não pode recuperar as informações de associação ao grupo do usuário de instalação. Nesse caso, o Assistente Preparar WebSphere MQ sempre solicita ao usuário os detalhes da conta do usuário do domínio para o serviço IBM WebSphere MQ Windows usar.

Quando o serviço IBM WebSphere MQ Windows precisar usar uma conta do usuário do domínio, o WebSphere MQ não poderá operar corretamente até que isso tenha sido configurado usando o Assistente Preparar WebSphere MQ . O Assistente Preparar WebSphere MQ não permite que o usuário continue com outras tarefas, até que o serviço Windows tenha sido configurado com uma conta adequada.

Se um domínio do Windows 2000 foi configurado com permissões de segurança não padrão, a solução usual para ativar o WebSphere MQ para funcionar corretamente é configurá-lo com uma conta do usuário do domínio adequada, conforme descrito acima.

Consulte Criando e configurando contas de domínio para o WebSphere MQ para obter mais informações

Configurando o IBM WebSphere MQ Services para executar sob um usuário do domínio no Windows
Use o assistente Preparar o IBM WebSphere MQ para inserir os detalhes da conta da conta do usuário do domínio. Como alternativa, é possível usar o painel Gerenciamento do Computador para alterar os detalhes de **Logon** para o serviço IBM WebSphere MQ específico da instalação.

Para obter mais informações, consulte Alterando a senha da conta de usuário do serviço do IBM WebSphere MQ Windows

Aplicando arquivos de modelo de segurança no Windows

Aplicar um modelo pode afetar as configurações de segurança aplicadas aos arquivos e diretórios do WebSphere MQ . Se você usar o modelo altamente seguro, aplique-o antes de instalar o WebSphere MQ

O Windows suporta arquivos de modelo de segurança baseados em texto que podem ser usados para aplicar configurações de segurança uniformes em um ou mais computadores com o snap-in do Security Configuration and Analysis MMC. Em particular, o Windows fornece vários modelos que incluem uma variedade de configurações de segurança com o objetivo de fornecer níveis específicos de segurança Esses modelos incluem Compatível, Seguro e Altamente Seguro.

Aplicar um desses modelos pode afetar as configurações de segurança aplicadas aos arquivos e diretórios do WebSphere MQ . Se desejar usar o modelo altamente seguro, configure sua máquina antes de instalar o WebSphere MQ.

Se você aplicar o modelo altamente seguro para uma máquina na qual o WebSphere MQ já está instalada, todas as permissões que você configurou nos arquivos e diretórios do WebSphere MQ serão removidas. Com a remoção dessas permissões, você perde acesso ao grupo de *Administradores*, *mqm* e, quando aplicável, ao grupo *Todos*, a partir dos diretórios de erro.

Grupos aninhados

Há restrições no uso de grupos aninhados. Eles resultam em parte do nível funcional do domínio e em parte das restrições do WebSphere MQ .

O Active Directory pode suportar diferentes tipos de grupos em um contexto de domínio, dependendo do nível funcional do domínio. Por padrão, os domínios do Windows 2003 estão no nível funcional *misto do Windows 2000* (Windows Server 2003, Windows XP, Windows Vista e Windows Server 2008 seguem o modelo de domínio do Windows 2003.) O nível funcional do domínio determina os tipos de grupos suportados e o nível de aninhamento permitido ao configurar IDs do usuário e um ambiente de domínio. Consulte a documentação do Active Directory para obter detalhes sobre o Escopo de Grupo e os critérios de inclusão.

Além dos requisitos do Active Directory , restrições adicionais são impostas aos IDs usados pelo WebSphere MQ As APIs de rede usadas pelo WebSphere MQ não suportam todas as configurações suportadas pelo nível funcional do domínio. Como resultado, o WebSphere MQ não é capaz de consultar as associações ao grupo de quaisquer IDs de Domínio presentes em um grupo Local de Domínio que é, então, aninhado em um grupo local. Além disso, o aninhamento múltiplo de grupos globais e universais não é suportado. No entanto, grupos globais e universais imediatamente aninhados são suportados.

Configurando autoridade adicional para aplicativos Windows se conectando ao IBM WebSphere MQ

A conta sob a qual as execuções de processos do IBM WebSphere MQ podem precisar de autorização adicional antes que o acesso SYNCHRONIZE aos processos do aplicativo possa ser concedido.

Você pode ter problemas se tiver aplicativos Windows , por exemplo, páginas ASP, conectando-se ao IBM WebSphere MQ que estão configurados para executar em um nível de segurança mais alto do que o usual

O IBM WebSphere MQ requer o acesso SYNCHRONIZE aos processos do aplicativo para coordenar determinadas ações. A APAR IC35116 mudou o IBM WebSphere MQ para que os privilégios apropriados sejam especificados. No entanto, a conta sob a qual as execuções de processo do IBM WebSphere MQ talvez precise de autorização adicional antes que o acesso solicitado seja concedido.

Quando um aplicativo do servidor tentar se conectar a um gerenciador de filas pela primeira vez, o IBM WebSphere MQ modificará o processo para conceder autoridade SYNCHRONIZE para os administradores do IBM WebSphere MQ. Para configurar autoridade adicional para o ID do usuário sob o qual os processos do IBM WebSphere MQ estão em execução, conclua as seguintes etapas:

1. Inicie a ferramenta Política de Segurança Local, clique em Configurações de Segurança-> Políticas Locais-> Atribuições de Direitos de Usuário, clique em "Depurar Programas"...
2. Dê um clique duplo em "Programas de depuração", em seguida, inclua seu ID do usuário do IBM WebSphere MQ na lista

Se o sistema estiver em um domínio do Windows e a configuração de política efetiva ainda não estiver configurada, mesmo que a configuração de política local esteja configurada, o ID do usuário deverá ser autorizado da mesma maneira no nível de domínio, usando a ferramenta Política de segurança de domínio...

Configurando segurança no HP Integrity NonStop Server

Considerações de segurança específicas para sistemas HP Integrity NonStop Server .

O cliente IBM WebSphere MQ para HP Integrity NonStop Server suporta os protocolos Transport Layer Security (TLS) e o Secure Sockets Layer (SSL) protocolos para fornecer segurança em nível de link quando você estiver se conectando a um gerenciador de filas. Esses protocolos são suportados usando uma implementação de OpenSSL. OpenSSL requer uma origem de dados aleatórios, para fornecer operações criptográficas fortes.

OpenSSL

OpenSSL visão geral de segurança para IBM WebSphere MQ client for HP Integrity NonStop Server.

O kit de ferramentas OpenSSL é uma implementação de origem aberta dos protocolos SSL (Secure Sockets Layer) e TLS (Transport Layer Security) para comunicações seguras sobre uma rede.

O kit de ferramentas é desenvolvido pelo Projeto OpenSSL. Para obter mais informações sobre o Projeto OpenSSL, consulte https://www.openssl.org. O cliente de IBM WebSphere MQ para o HP Integrity NonStop Server contém versões modificadas das bibliotecas de OpenSSL e o comando **openssl**. As bibliotecas e o comando **openssl** são transportados do kit de OpenSSL 1.0.1c, e são fornecidos apenas como código de objeto. Nenhum código fonte é fornecido.

As bibliotecas OpenSSL são carregadas por programas de aplicativo IBM WebSphere MQ do cliente dinamicamente conforme necessário. Apenas as bibliotecas OpenSSL, fornecidas pelo IBM WebSphere MQ, são suportadas para uso com aplicativos cliente IBM WebSphere MQ.

O comando **openssl** , que pode ser usado para fins de gerenciamento de certificado, é instalado no diretório OSS `opt_installation_path/opt/mqm/bin`.

Usando o comando **openssl** , você pode criar e gerenciar chaves e certificados digitais com diversos formatos de dados comuns, e realizar tarefas simples de autoridade de certificação (CA).

O formato padrão para chave e dados do certificado que é processado pelo OpenSSL é o formato Privacy Enhanced Mail (PEM). Dados no formato PEM será base64 codificada dados ASCII. Os dados, portanto, podem ser transferidos usando sistemas baseados em texto, como email, e podem ser cortados e colados utilizando editores de texto e navegadores da web. PEM é um padrão Internet para trocas de criptografia em texto e está especificado na Internet RFCs 1421, 1422, 1423, e 1424. O IBM WebSphere MQ assume que um arquivo com extensão `.pem` contém dados no formato PEM Um arquivo no formato PEM pode conter vários certificados e outros objetos codificados e pode incluir comentários.

O suporte de SSL do IBM WebSphere MQ em outros sistemas operacionais pode requerer dados de chave e de certificado em arquivos que serão codificados usando o Distinguished Encoding Rules (DER). DER é um conjunto de regras de codificação para usar a notação ASN.1 em comunicações seguras. Dados que são codificados utilizando DER são dados binários e o formato dos dados de certificado e de chave que é codificado utilizando DER também é conhecido como PKCS#12 ou PFX. Um arquivo que contém esses dados geralmente tem uma extensão de `.p12` ou `.pfx`. O comando **openssl** pode converter entre o formato PEM e PKCS#12.

Entropia do Daemon

OpenSSL requer uma origem de dados aleatórios, para fornecer operações criptográficas fortes. Geração de número aleatório é um recurso geralmente fornecido pelo sistema operacional ou por um processo daemon de todo o sistema. O HP Integrity NonStop Server sistema operacional não oferecem esse recurso no sistema operacional.

Quando estiver usando o suporte de SSL e TLS fornecido com o cliente de IBM WebSphere MQ para o HP Integrity NonStop Server, um processo que é chamado de um daemon de entropia é necessário para

fornecer a origem dos dados aleatórios. Quando você inicia um canal do cliente que requer SSL ou TLS, o OpenSSL espera que um daemon de entropia esteja executando e fornecendo seus serviços em um soquete no sistema de arquivos OSS em `/etc/egd-pool`.

Um daemon de entropia não é fornecido pelo cliente IBM WebSphere MQ para HP Integrity NonStop Server.. O cliente IBM WebSphere MQ para HP Integrity NonStop Server é testado com os daemons de entropia a seguir:

- amqjkdm0 (como fornecida pelo IBM WebSphere MQ servidor 5,3)
- `/usr/local/bin/prngd` (Version 0.9.27, conforme fornecido pelo HP Integrity NonStop Server Open Source Technical Library)

Configurando a segurança do cliente MQI do IBM WebSphere MQ

Você deve considerar a segurança do cliente MQI do IBM WebSphere MQ para que os aplicativos clientes não tenham acesso irrestrito a recursos no servidor.

Ao executar um aplicativo cliente, não execute o aplicativo usando um ID do usuário que tenha mais direitos de acesso do que necessário; por exemplo, um usuário no grupo `mqm` ou até mesmo o usuário `mqm` em si.

Ao executar um aplicativo como um usuário com direitos de acesso em excesso, você corre o risco de o aplicativo acessar e alterar as partes do gerenciador de filas, seja por acaso ou intencionalmente.

Existem dois aspectos de segurança entre um aplicativo cliente e seu servidor do gerenciador de filas: autenticação e controle de acesso.

- A autenticação pode ser usada para assegurar que o aplicativo cliente, em execução como um usuário específico, é quem eles dizem que são. Ao usar autenticação é possível evitar que um invasor ganhe acesso ao seu gerenciador de filas personificando um de seus aplicativos.

  Você deve usar a autenticação mútua em SSL ou TLS. Para obter informações adicionais, consulte “Trabalhando com SSL ou TLS” na página 113
- O controle de acesso pode ser usado para conceder ou remover direitos de acesso para um usuário ou grupo específico de usuários. Ao executar um aplicativo cliente com um usuário especificamente criado (ou usuário em um grupo específico), é possível usar controles de acesso para assegurar que o aplicativo não possa acessar partes de seu gerenciador de filas que ele não deve.

  Ao configurar o controle de acesso, deve-se considerar as regras de autenticação de canal e o campo MCAUSER em um canal. Ambos os recursos têm a capacidade de mudar o ID do usuário que está sendo usado para verificar direitos de controle de acesso.

  Para obter informações adicionais sobre controle de acesso, consulte o “Autorizando o acesso aos objetos” na página 162.

Se você configurou um aplicativo cliente para se conectar a um canal específico com um ID restrito, mas o canal tem um ID de administrador configurado em seu campo MCAUSER, então, considerando que o aplicativo cliente se conecte com sucesso, o ID de administrador é usado para verificações do controle de acesso. Portanto, o aplicativo cliente terá direitos de acesso completos ao seu gerenciador de filas.

Para obter mais informações sobre o atributo MCAUSER, consulte “Mapeando um ID do Usuário Decrarado pelo Cliente para um ID do Usuário MCAUSER” na página 188.

As regras de autenticação de canal também podem ser usadas como um método para controlar o acesso a um gerenciador de filas, configurando regras e os critérios específicos para uma conexão para ser aceito.

Para obter mais informações sobre as regras de autenticação de canal, consulte: “Registros de Autenticação de Canal” na página 40.

Especificando que Apenas CipherSpecs Certificados por FIPS São Usados no Tempo de Execução no Cliente de MQI

Crie seus repositórios de chaves usando software compatível com FIPS e, em seguida, especifique que o canal deve usar CipherSpecs certificados por FIPS.

Para serem compatíveis com FIPS no tempo de execução, os repositórios de chaves devem ter sido criados e gerenciados apenas com o uso de software compatível com FIPS, como runmqakm com a opção -fips.

É possível especificar que um canal SSL ou TLS deve usar apenas CipherSpecs certificados por FIPS de três maneiras, listadas por ordem de precedência:

1. Configure o campo FipsRequired na estrutura MQSCO como MQSSL_FIPS_YES.
2. Configure a variável de ambiente MQSSLFIPS como YES.
3. Configure o atributo SSLFipsRequired no arquivo de configuração do cliente como YES.

Por padrão, CipherSpecs certificados por FIPS não são obrigatórios.

Estes valores possuem o mesmo significado que os valores de parâmetro equivalentes em ALTER QMGR SSLFIPS (consulte ALTER QMGR). Se atualmente o processo do cliente não tiver conexões SSL ou TLS ativas, e um valor FipsRequired for especificado validamente em SSL MQCONNX, todas as conexões SSL subsequentes associadas a esse processo deverão usar apenas CipherSpecs associados a esse valor. Isso se aplica até que esta e todas as outras conexões SSL ou TLS sejam interrompidas, em cujo estágio um MQCONNX subsequente pode fornecer um novo valor para FipsRequired.

Se o hardware de criptografia estiver presente, os módulos criptográficos usados pelo WebSphere MQ poderão ser configurados para serem aqueles módulos fornecidos pelo produto de hardware e poderão ser certificados por FIPS para um nível específico Os módulos configuráveis, e se eles são certificados por FIPS, dependem do produto de hardware em uso.

Onde for possível, se CipherSpecs apenas FIPS forem configurados, o cliente de MQI rejeitará conexões que especifiquem um CipherSpec não FIPS com MQRC_SSL_INITIALIZATION_ERROR. WebSphere MQ não garante rejeitar todas essas conexões e é sua responsabilidade determinar se a configuração do WebSphere MQ é compatível com FIPS.

Conceitos relacionados

“Federal Information Processing Standards (FIPS) para UNIX, Linuxe Windows” na página 26
Quando a criptografia é necessária em um canal SSL ou TLS no Windows, UNIX and Linux sistemas, WebSphere MQ usa um pacote de criptografia chamado IBM Crypto for C (ICC). Nas plataformas Windows, UNIX and Linux , o software ICC passou pelo Federal Information Processing Standards (FIPS) Cryptomodule Validation Program do US National Institute of Standards and Technology, no nível 140-2.

Sub-rotina SSL do Arquivo de Configuração do Cliente

Referências relacionadas

FipsRequired (MQLONG)

MQSSLFIPS

Executando os aplicativos cliente SSL ou TLS com diversas instalações do GSKit V8.0 no AIX

Os aplicativos cliente SSL ou TLS no AIX podem experimentar os erros MQRC_CHANNEL_CONFIG_ERROR e AMQ6175 quando executados em sistemas AIX com várias instalações do GSKit V8.0.

Ao executar aplicativos cliente em um sistema AIX com várias instalações de GSKit V8.0, as chamadas de conexão do cliente podem retornar MQRC_CHANNEL_CONFIG_ERROR ao usar SSL ou TLS. Os logs do /var/mqm/errors registram os erros AMQ6175 e AMQ9220 para o aplicativo cliente com falha, por exemplo:

```
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 ----------------------------------------------------------
```

Uma causa comum desse erro é que a configuração da variável de ambiente LIBPATH ou LD_LIBRARY_PATH fez com que o cliente IBM WebSphere MQ carregasse um conjunto misto de bibliotecas de duas instalações do GSKit V8.0 diferentes. Executar um aplicativo clienteIBM WebSphere MQ em um ambiente Db2 pode causar esse erro.

Para evitar este erro, inclua os diretórios da biblioteca do IBM WebSphere MQ na frente do caminho da biblioteca para que as bibliotecas do IBM WebSphere MQ tenham precedência. Isso pode ser feito usando o comando **setmqenv** com o parâmetro **-k** , por exemplo:

```
. /usr/mqm/bin/setmqenv -s -k
```

Para obter mais informações sobre o uso do comando **setmqenv** , consulte setmqenv (configurar o ambiente WebSphere MQ)

Configurando comunicações para SSL ou TLS em sistemas UNIX, Linux, and Windows

As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

Para configurar sua instalação de SSL ou TLS, você deve definir seus canais para usar SSL ou TLS. Você deve também criar e gerenciar seus certificados digitais. Nos sistemas UNIX, Linux e Windows , é possível executar testes com certificados autoassinados.

Certificados autoassinados não podem ser revogados, isso poderia permitir que um invasor copiasse uma identidade após o comprometimento de uma chave privada. CAs podem revogar um certificado comprometido, evitando seu uso adicional. O uso de certificados assinados por CA são, portanto, mais seguros em um ambiente de produção, embora certificados autoassinados sejam mais convenientes em um sistema de teste.

Para obter informações completas sobre a criação e o gerenciamento de certificados, consulte “Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows” na página 116.

Esta coleção de tópicos apresenta algumas das tarefas envolvidas na configuração de comunicações SSL, e fornece orientação passo a passo para concluir essas tarefas.

Você também pode querer testar a autenticação de cliente SSL ou TLS, que são uma parte opcional dos protocolos. Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação do IBM WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente.

Em sistemas UNIX, Linux e Windows, o cliente SSL ou TLS enviará um certificado somente se tiver um com rótulo no formato correto do IBM WebSphere MQ:

- Para um gerenciador de filas, o formato é `ibmwebspheremq`, seguido pelo nome de seu gerenciador de filas mudado para minúsculas. Por exemplo, para `QM1`, `ibmwebspheremqqm1`
- Para um cliente IBM WebSphere MQ, `ibmwebspheremq` seguido por seu ID do usuário de logon mudado para minúsculas, por exemplo, `ibmwebspheremqmyuserid`.

O IBM WebSphere MQ usa o prefixo `ibmwebspheremq` em um rótulo para evitar confusão com certificados de outros produtos. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falhará somente se a extremidade do canal que age como o servidor SSL ou TLS estiver definida com o parâmetro SSLCAUTH configurado como REQUIRED ou um valor de parâmetro SSLPEER configurado. Para obter informações adicionais, consulte “Conectando dois gerenciadores de filas usando SSL ou TLS” na página 210.

Trabalhando com SSL ou TLS

Esses tópicos fornecem instruções para executar tarefas únicas relacionadas ao uso de SSL ou TLS com o IBM WebSphere MQ.

Muitos deles são usados como etapas nas tarefas de alto nível descritos nas seguintes seções:

- “Identificando e autenticando usuários” na página 147
- “Autorizando o acesso aos objetos” na página 162
- “Confidencialidade das mensagens” na página 209
- “Integridade de dados de mensagens” na página 230
- “Mantendo Clusters Seguros” na página 249

Trabalhando com SSL ou TLS no HP Integrity NonStop Server

Descreve o cliente de IBM WebSphere MQ para a implementação da segurança OpenSSL do HP Integrity NonStop Server, incluindo serviços de segurança, componentes, versões de protocolo suportadas, CipherSpecs suportadas e funcionalidade de segurança não suportada.

IBM WebSphere MQ O suporte SSL e TLS fornece os seguintes serviços de segurança para canais do cliente:

- Autenticação do servidor e, opcionalmente, autenticação do cliente.
- Criptografia e decriptografia dos dados que estão fluindo por um canal.
- Verificações de integridade nos dados que estão fluindo em um canal.

O suporte de SSL e TLS fornecido com o cliente de IBM WebSphere MQ para o HP Integrity NonStop Server abrange os seguintes componentes:

- Bibliotecas de OpenSSL e o comando **`openssl`**.
- senha stash de comandos doIBM WebSphere MQ , **`amqrsslc`**.

Os seguintes componentes requeridos para operação de canal de cliente SSL ou TLS não são fornecidos com o cliente de IBM WebSphere MQ para o HP Integrity NonStop Server:

- Um daemon de entropia para fornecer uma fonte de dados aleatórios para criptografia OpenSSL.

Versões de Protocolo Suportadas

O cliente IBM WebSphere MQ para HP Integrity NonStop Server suporta as versões de protocolos a seguir:

- SSL 3.0
- TLS 1.0
- TLS 1.2

CipherSpecs Suportados

O cliente IBM WebSphere MQ para HP Integrity NonStop Server suporta as versões de CipherSpecs a seguir:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- RC4_SHA_US
- RC4_MD5_US
- TRIPLE_DES_SHA_US
- TLS_RSA_WITH_3DES_EDE_CBC_SHA (descontinuado)
- DES_SHA_EXPORT1024
- RC4_56_SHA_EXPORT1024
- RC4_MD5_EXPORT
- RC2_MD5_EXPORT
- DES_SHA_EXPORT
- TLS_RSA_WITH_DES_CBC_SHA
- NULL_SHA
- NULL_MD5
- FIPS_WITH_DES_CBC_SHA
- FIPS_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_NULL_SHA256
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- ECDHE_ECDSA_AES_128_CBC_SHA256
- ECDHE_ECDSA_AES_256_CBC_SHA384
- ECDHE_RSA_AES_128_CBC_SHA256
- ECDHE_RSA_AES_256_CBC_SHA384
- ECDHE_ECDSA_AES_128_GCM_SHA256
- ECDHE_ECDSA_AES_256_GCM_SHA384
- ECDHE_RSA_AES_128_GCM_SHA256
- ECDHE_RSA_AES_256_GCM_SHA384

Funcionalidade de Segurança não Suportada

O cliente IBM WebSphere MQ para HP Integrity NonStop Server não suporta atualmente:

- PKCS#11 Suporte de hardware criptográfico

- a verificação da lista de Revogação de Certificado LDAP
- verificação OCSP Online Certificate Status Protocol
- FIPS 140-2, controles do conjunto de cifras do CONJUNTO B da NSA

Gerenciamento de certificado

Use um conjunto de arquivos para o armazenamento de certificados digitais e informações de revogação de certificado.

O suporte de SSL e TLS do IBM WebSphere MQ usa um conjunto de arquivos para armazenar informações de certificado digital e de revogação de certificado. Esses arquivos estão localizados em um diretório especificado, programaticamente por meio do campo KeyRepository na estrutura MQSCO passada na chamada MQCONNX, pela variável de ambiente *MQSSLKEYR* variável de ambiente, ou, na sub-rotina do SSL do `mqclient.ini` utilizando o atributo SSLKeyRepository.

A estrutura MQSCO tem precedência sobre a variável de ambiente MQSSLKEYR que tem precedência sobre o valor da sub-rotina do arquivo ini.

Importante: O local do repositório de chaves especifica um local de diretório e não um nome de arquivo na plataforma do HP Integrity NonStop Server.

O cliente de IBM WebSphere MQ para o HP Integrity NonStop Server usa os seguintes arquivos nomeados, que fazem distinção entre maiúsculas e minúsculas, no local do repositório de chaves:

- "Armazenamento de Certificados Pessoais" na página 115
- "Armazenamento confiável de certificado" na página 115
- "Arquivo Stash de Passphrase" na página 116
- "Arquivo de Lista de Revogação de Certificado" na página 116

Armazenamento de Certificados Pessoais
O arquivo de armazenamento de certificados pessoais, `cert.pem`.

Esse arquivo contém o certificado pessoal e a chave privada criptografada para o cliente usar, no formato PEM. A existência desse arquivo é opcional quando você estiver usando canais SSL ou TLS que não requerem autenticação de cliente. Se a autenticação de cliente for necessária pelo canal, e SSLCAUTH(REQUIRED) seja especificado na definição de canal, esse arquivo deve existir e conter o certificado e a chave privada criptografada.

Permissões de arquivo devem ser configuradas neste arquivo para permitir acesso de leitura para o proprietário do armazenamento de certificados.

Um arquivo corretamente formatado `cert.pem` deve conter exatamente duas seções com os seguintes cabeçalhos e rodapés:

```
-----BEGIN PRIVATE KEY-----
Base 64 ASCII encoded private key data here
-----END PRIVATE KEY-----
```

```
-----BEGIN CERTIFICATE-----
Base 64 ASCII encoded certificate data here
-----END CERTIFICATE-----
```

A frase de passagem para a chave privada criptografada é armazenada no arquivo stash frase de passagem, `Stash.sth`.

Armazenamento confiável de certificado
O arquivo de armazenamento de certificado, `trust.pem`.

Esse arquivo contém os certificados que são necessários para validar os certificados pessoais que são usados pelos gerenciadores de filas ao qual o cliente se conecta, no formato PEM. O armazenamento confiável de certificados é obrigatório para todos os canais de cliente SSL ou TLS.

Permissões de arquivo devem ser configuradas para limitar acesso de gravação a esse arquivo.

Um arquivo `trust.pem` corretamente formatado deve conter uma ou mais seções com os seguintes cabeçalhos e rodapés:

```
-----BEGIN CERTIFICATE-----
Base 64 ASCII encoded certificate data here
-----END CERTIFICATE-----
```

Arquivo Stash de Passphrase
O arquivo stash passphrase, `Stash.sth`.

Este arquivo é um formato binário privado ao IBM WebSphere MQ e contém a frase de senha criptografada para utilização quando você estiver acessando a chave privada que é mantida no arquivo `cert.pem` . A chave privada em si é armazenado no armazenamento de certificados `cert.pem` .

Este arquivo é criado ou alterado usando a ferramenta de linha de comandos do IBM WebSphere MQ **amqrsslc** com o parâmetro **-s**. Por exemplo, onde o diretório `/home/alice` contém um `cert.pem` arquivo:

```
amqrsslc -s /home/alice/cert

  Enter password for Keystore /home/alice/cert.pem :
    password

  Stashed the password in file /home/alice/Stash.sth
```

As permissões de arquivo devem ser definidas nesse arquivo para permitir o acesso de leitura ao proprietário do armazenamento de certificados pessoais associados.

Arquivo de Lista de Revogação de Certificado
O arquivo de lista de revogação de certificado, `crl.pem`.

Esse arquivo contém as listas de revogação de certificado (CRLs) que o cliente utiliza para validar certificados digitais, em formato PEM. A existência desse arquivo é opcional. Se esse arquivo não estiver presente, nenhuma verificação de revogação de certificado são feitas quando você estiver validando os certificados.

Permissões de arquivo devem ser configuradas para limitar acesso de gravação a esse arquivo.

Um arquivo corretamente formatado `crl.pem` deve conter uma ou mais seções com os seguintes cabeçalhos e rodapés:

```
-----BEGIN X509 CRL-----
Base 64 ASCII encoded CRL data here
-----END X509 CRL-----
```

Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows

Nos sistemas UNIX, Linux e Windows , o suporte Secure Sockets Layer (SSL) é instalado com IBM WebSphere MQ.

Para obter informações mais detalhadas sobre as políticas de validação de certificado, consulte Validação do certificado e design da política de confiança.

Usando iKeyman, iKeycmd, runmqakm e runmqckm

Nos sistemas UNIX, Linux e Windows , gerencie chaves e certificados digitais com a GUI do iKeyman ou a partir da linha de comandos usando iKeycmd ou runmqakm.

- Para sistemas **UNIX and Linux** :
  - Use o comando **strmqikm** para iniciar a GUI iKeyman.
  - Use o comando **runmqckm** para executar tarefas com a interface da linha de comandos do iKeycmd.
  - Use o comando **runmqakm** para executar tarefas com a interface da linha de comando runmqakm. A sintaxe de comando para **runmqakm** é a mesma que a sintaxe para **runmqckm**.

Se for preciso gerenciar certificados SSL de um modo compatível com FIPS, use o comando **runmqakm** no lugar dos comandos **runmqckm** ou **strmqikm**.

Consulte Gerenciando chaves e certificados para uma descrição completa das interfaces da linha de comandos para os comandos **runmqckm** e **runmqakm**.

Se você estiver usando certificados ou chaves armazenadas no hardware de criptografia PKCS #11, observe que iKeycmd e iKeyman são programas de 64 bits. Módulos externos necessários para o suporte do PKCS #11 serão carregados em um processo de 64 bits, portanto, você deverá ter uma biblioteca do PKCS #11 de 64 bits instalada para a administração do hardware de criptografia. As plataformas Windows e Linux x86 de 32 bits são as únicas exceções, pois os programas iKeyman e iKeycmd são de 32 bits nessas plataformas.

Nas plataformas a seguir, em que o JRE era de 32 bits em versões anteriores do produto, mas é de 64 bits apenas no IBM WebSphere MQ Version 7.5, pode ser necessário instalar drivers PKCS#11 adicionais apropriados para o modo de endereçamento do **iKeyman** e **iKeycmd** JRE. Isso acontece porque o driver PKCS#11 deve usar o mesmo modo de endereçamento do JRE. A tabela a seguir mostra os modos de endereçamento JRE do IBM WebSphere MQ Version 7.5.

Tabela 12. Modos de endereçamento do JRE IBM WebSphere MQ Version 7.5

Plataforma	**Modo de Endereçamento JRE**
Windows (32 bits ou 64 bits)	32
Linux para System x 32 bits	32
Linux para System x 64 bits	64
Linux para System p	64
Linux para System z	64
HP-UX	64
Solaris SPARC	64
Solaris x86-64	64
AIX	64

Antes de executar o comando **strmqikm** para iniciar a GUI do iKeyman, assegure-se de estar trabalhando em uma máquina que esteja apta para executar o X Window System e faça o seguinte:

– Defina a variável de ambiente DISPLAY, por exemplo:

```
export DISPLAY=mypc:0
```

– Verifique se a variável de ambiente PATH contém **/usr/bin** e **/bin**. Isso também é necessário para os comandos **runmqckm** e **runmqakm** Por exemplo:

```
export PATH=$PATH:/usr/bin:/bin
```

- Para sistemas **Windows** :
  – Use o comando **strmqikm** para iniciar a GUI iKeyman.
  – Use o comando **runmqckm** para executar tarefas com a interface da linha de comandos do iKeycmd.

    Se for preciso gerenciar certificados SSL de um modo compatível com FIPS, use o comando **runmqakm** no lugar dos comandos **runmqckm** ou **strmqikm**.

Para solicitar o rastreio de SSL nos sistemas UNIX, Linux ou Windows , consulte strmqtrc

Referências relacionadas
Comandos runmqckm e runmqakm

Configurando um repositório de chaves nos sistemas UNIX, Linux, and Windows

É possível configurar um repositório de chaves usando a interface com o usuário iKeyman ou usando os comandos **iKeycmd** ou **runmqakm** .

Sobre esta tarefa

Uma conexão SSL ou TLS requer um *repositório de chaves* em cada extremidade da conexão. Cada gerenciador de filas do IBM WebSphere MQ e IBM WebSphere MQ MQI client devem ter acesso a um repositório de chaves Para obter mais informações, consulte "O repositório de chaves SSL ou TLS" na página 24.

Em sistemas UNIX, Linux, and Windows, os certificados digitais são armazenados em um arquivo do banco de dados de chave que é gerenciado usando a interface com o usuário **iKeyman** ou usando os comandos **iKeycmd** ou **runmqakm**. Esses certificados digitais têm rótulos. Um rótulo específico associa um certificado pessoal a um gerenciador de filas ou IBM WebSphere MQ MQI client. SSL e TLS usam este certificado com a finalidade de autenticação. Em sistemas UNIX, Linux, and Windows , o IBM WebSphere MQ usa `ibmwebspheremq` como um prefixo de rótulo para evitar confusão com certificados para outros produtos.. O prefixo é seguido pelo nome do gerenciador de filas ou ID de logon do usuário do IBM WebSphere MQ MQI client , alterado para minúsculas. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O nome do arquivo do banco de dados de chave contém um caminho e nome de raiz:

- Em sistemas UNIX and Linux, o caminho padrão para um gerenciador de filas (configurado durante a criação do gerenciador de filas) é `/var/mqm/qmgrs/<queue_manager_name>/ssl`.

  Em sistemas Windows , o caminho padrão é *MQ_INSTALLATION_PATH*`\Qmgrs\`*queue_manager_name*`\ssl`, em que *MQ_INSTALLATION_PATH* é o diretório no qual o IBM WebSphere MQ está instalado Por exemplo, `C:\program files\IBM\WebSphere MQ\Qmgrs\QM1\ssl`.

  O nome de raiz padrão é `key`. Como opção, você pode escolher seu próprio nome de caminho e de stem, mas a extensão deve ser `.kdb`.

  Se você escolher seu próprio caminho ou nome de arquivo, configure as permissões para que o arquivo controle rigorosamente o acesso a ele.
- Para um cliente WebSphere MQ , não há caminho padrão ou nome de raiz. Controle rigorosamente o acesso a esse arquivo. A extensão deve ser `.kdb`.

Não crie os repositórios de chaves em um sistema de arquivos que não suporta bloqueios no nível de arquivos, por exemplo, o NFS versão 2 em sistemas Linux.

Consulte "Mudando o local do repositório de chaves para um gerenciador de filas em sistemas UNIX, Linux ou Windows" na página 122 para obter informações sobre a verificação e a especificação do nome do arquivo de banco de dados de chave. Você pode especificar o nome do arquivo de banco de dados de chave antes ou depois de criar o arquivo de banco de dados de chave.

O ID do usuário a partir do qual você executa o comando **iKeyman** ou **iKeycmd** deve ter permissão de gravação para o diretório no qual o arquivo de banco de dados de chave foi criado ou atualizado. Para um gerenciador de filas usando o diretório padrão `ssl`, o ID do usuário a partir do qual você executa o **iKeyman** ou **iKeycmd** deve ser um membro do grupo mqm. Para um IBM WebSphere MQ MQI client, se você executar **iKeyman** ou **iKeycmd** a partir de um ID do usuário diferente daquele sob o qual o cliente é executado, deverá alterar as permissões de arquivo para ativar o IBM WebSphere MQ MQI client para acessar o arquivo de banco de dados de chave no tempo de execução. Para obter mais informações, consulte "Acessando e protegendo seus arquivos de banco de dados de chaves no Windows" na página 120 ou "Acessando e Protegendo seus Arquivos do Banco de Dados de Chaves nos Sistemas UNIX and Linux" na página 120.

No **iKeyman** ou **iKeycmd** versão 7.0, novos bancos de dados de chave são automaticamente preenchidos com um conjunto de certificados de autoridade de certificação (CA) predefinidos. No **iKeyman** ou **iKeycmd** versão 8.0, os bancos de dados de chaves não são preenchidos automaticamente,

tornando a configuração inicial mais segura porque você inclui apenas os certificados de autoridade de certificação desejados, em seu arquivo de banco de dados de chave

Nota: Devido a essa mudança no comportamento para o GSKit versão 8.0 que resulta em certificados de autoridade de certificação não serem mais incluídos automaticamente no repositório, deve-se incluir manualmente seus certificados de autoridade de certificação preferenciais. Essa mudança de comportamento fornece a você um controle mais granular sobre os certificados de CA usados. Consulte o “Incluindo certificados de autoridade de certificação padrão em um repositório de chaves vazio em sistemas UNIX, Linux, and Windows com GSKit versão 8.0” na página 121.

Você cria o banco de dados de chaves usando a linha de comandos ou usando a interface com o usuário **strmqikm** (iKeyman).

Nota: Se deve-se gerenciar certificados TLS de uma maneira que esteja de acordo com FIPS, use o comando **runmqakm**. A interface com o usuário **strmqikm** não fornece uma opção compatível com FIPS.

Procedimento

Crie um banco de dados de chave usando a linha de comandos

1. Execute um dos comandos a seguir:
   - Nos sistemas UNIX, Linux, and Windows:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Usando runmqakm:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
         -stash -fips -strong
     ```

   em que:

   -db *filename*
   Especifica o nome completo do arquivo de um banco de dados de chaves do CMS e deve ter uma extensão de arquivo de `.kdb`.

   -pw *password*
   Especifica a senha para o banco de dados de chaves do CMS.

   -type *cms*
   Especifica o tipo de banco de dados. (Para IBM WebSphere MQ, ele deve ser `cms`.)

   -stash
   Salva a senha do banco de dados de chaves em um arquivo.

   -fips
   Desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando em modo FIPS, o componente ICC usa algoritmos que são validados para FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

   -strong
   Verifica se a senha inserida atende aos requisitos mínimos de força da senha. Os requisitos mínimos para uma senha são como a seguir:

   - A senha deve ter no mínimo 14 caracteres.
   - A senha deve conter no mínimo um caractere minúsculo, um caractere maiúsculo e um dígito ou caractere especial. Os caracteres especiais incluem o asterisco (*), o sinal de dólar ($), o sinal de número (#) e o sinal de percentual (%). Um espaço é classificado como um caractere especial.
   - Cada caractere pode ocorrer no máximo de três vezes em uma senha.
   - O máximo de dois caracteres consecutivos na senha podem ser idênticos.
   - Todos os caracteres estão no conjunto de caracteres padrão ASCII para impressão no intervalo 0x20 - 0x7E.

Como alternativa, crie um banco de dados de chaves usando a interface com o usuário **strmqikm** (iKeyman).

2. Nos sistemas UNIX and Linux, efetue login como usuário raiz. Nos sistemas Windows, efetue login como Administrador ou como um membro do grupo MQM.
3. Inicie a interface com o usuário do iKeyman executando o comando **strmqikm**.
4. No menu **Arquivo do Banco de Dados de Chave**, clique em **Novo**.

   A janela Novo é aberta.
5. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
6. No campo **Nome do Arquivo**, digite um nome de arquivo.

   Esse campo já contém o texto `key.kdb`. Se o seu nome de raiz for `key`, deixe este campo inalterado. Se você especificou um nome de raiz diferente, substitua `key` pelo seu nome de raiz. No entanto, não se deve mudar a extensão `.kdb`.
7. No campo **Localização**, digite o caminho.

   Por exemplo:

   - Para um gerenciador de filas: `/var/mqm/qmgrs/QM1/ssl` (em sistemas UNIX and Linux) ou `C:\Program Files\IBM\WebSphere MQ\qmgrs\QM1\ssl` (em sistemas Windows)

     O caminho deve corresponder ao valor do atributo **SSLKeyRepository** do gerenciador de filas.
   - Para um cliente IBM WebSphere MQ: `/var/mqm/ssl` (em sistemas UNIX and Linux) ou `C:\mqm\ssl` (em sistemas Windows).
8. Click **Open**.

   A janela Prompt de Senha é aberta.
9. Digite a senha no campo **Senha** e digite-a novamente no campo **Confirmar Senha**.
10. Selecione a caixa de seleção **Stash the password to a file**.

    Nota: Se você não proteger a senha, as tentativas para iniciar os canais SSL ou TLS falharão porque não será possível obter a senha necessária para acessar o arquivo do banco de dados de chave.
11. Clique em **OK**.

    A janela Certificados pessoais é aberta.
12. Configure as permissões de acesso conforme descrito em "Acessando e protegendo seus arquivos de banco de dados de chaves no Windows" na página 120 ou "Acessando e Protegendo seus Arquivos do Banco de Dados de Chaves nos Sistemas UNIX and Linux" na página 120.

Acessando e protegendo seus arquivos de banco de dados de chaves no Windows
Os arquivos do banco de dados de chave podem não ter as permissões de acesso apropriadas. Você deve configurar o acesso apropriado a esses arquivos.

Configure o controle de acesso para os arquivos *key*`.kdb`, *key*`.sth`, *key*`.crl` e *key*`.rdb`, em que *key* é o nome de raiz do seu banco de dados de chaves, para conceder autoridade para um conjunto restrito de usuários.

Considere conceder acesso da seguinte forma:

autoridade plena
BUILTIN\Administrators, NT AUTHORITY\SYSTEM e o usuário que criou os arquivos de banco de dados.

autoridade de leitura
Para um gerenciador de filas, apenas o grupo mqm local. Isto supõe que o MCA esteja em execução com um ID do usuário no grupo mqm.

Para um cliente, o ID do usuário com o qual o processo do cliente está sendo executado.

Acessando e Protegendo seus Arquivos do Banco de Dados de Chaves nos Sistemas UNIX and Linux
Os arquivos do banco de dados de chave podem não ter as permissões de acesso apropriadas. Você deve configurar o acesso apropriado a esses arquivos.

Para um gerenciador de filas, configure permissões nos arquivos de banco de dados de chave para que os processos de gerenciador de filas e de canal possam lê-los quando necessário, mas outros usuários não possam ler ou modificá-los. Normalmente, o usuário mqm precisa de permissões de leitura. Se você tiver criado o arquivo do banco de dados de chave efetuando login como o usuário mqm, as permissões serão provavelmente suficientes; se você não era o usuário mqm, mas outro usuário nesse grupo, provavelmente precisará conceder permissões de leitura a outros usuários no grupo mqm.

De forma semelhante para um cliente, configure permissões nos arquivos de banco de dados de chave para que os processos de cliente possam lê-los quando necessário, mas outros usuários não possam ler ou modificá-los. Normalmente, o usuário sob o qual o processo do cliente é executado precisa de permissões de leitura. Se você tiver criado o arquivo do banco de dados de chave efetuando login como esse usuário, as permissões serão provavelmente suficientes; se você não era o usuário do processo de cliente, mas outro usuário nesse grupo, provavelmente precisará conceder permissões de leitura a outros usuários no grupo.

Configure as permissões nos arquivos *key*.kdb, *key*.sth, *key*.crl e *key*.rdb, em que *key* é o nome de raiz do banco de dados de chave, para leitura e gravação para o proprietário do arquivo e para leitura para o grupo de usuários mqm ou cliente (-rw-r-----).

Incluindo certificados de autoridade de certificação padrão em um repositório de chaves vazio em sistemas UNIX, Linux, and Windows com GSKit versão 8.0
Siga este procedimento para incluir um ou mais dos certificados de CA padrão em um repositório de chaves vazio com GSKit versão 8.

No GSKit versão 7.0, o comportamento ao criar um novo repositório de chaves era incluir automaticamente em um conjunto de certificados de autoridade de certificação padrão para autoridades de certificação comumente usadas. Para o GSKit versão 8, esse comportamento foi alterado para que os certificados de CA não sejam mais incluídos automaticamente no repositório. Agora é exigido que o usuário inclua certificados de CA manualmente no repositório de chaves.

Usando o iKeyman

Execute as seguintes etapas na máquina na qual deseja incluir o certificado de CA:

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (nos sistemas UNIX Linux e Windows).
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo de banco de dados de chave no qual você quer incluir o certificado, por exemplo key.kdb.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do arquivo de banco de dados de chave é exibido no campo **Nome de Arquivo**.
8. No campo **Conteúdo do banco de dados de chave**, selecione **Certificados de Assinante**.
9. Clique em **Preencher**. A janela Incluir Certificado de CA é aberta.
10. Os certificados de CA que estão disponíveis para serem incluídos no repositório são exibidos em uma estrutura de árvore hierárquica. Selecione a entrada de nível superior para a organização em cujos certificados de CA você quer confiar para visualizar a lista completa de certificados de CA válidos.
11. Selecione na lista os certificados de CA nos quais deseja confiar e clique em **OK**. Os certificados são incluídos no repositório de chaves.

Usando a linha de comandos

Use os comandos a seguir para listar e, em seguida, incluir certificados de CA usando iKeycmd:

- Emita o seguinte comando para listar os certificados de CA padrão junto com as organizações que os emitem:

```
runmqckm -cert -listsigners
```

- Emita o seguinte comando para incluir todos os certificados de CA para a organização especificada no campo *rótulo*:

```
runmqckm -cert -populate -db filename -pw password -label label
```

em que:

-db *filename*	é o nome do caminho completo do banco de dados de chaves.
-pw *password*	é a senha do banco de dados de chave.
-label *label*	é o rótulo anexado ao certificado.

Nota: Incluir um certificado de autoridade de certificação em um repositório de chaves resulta em WebSphere MQ confiar em todos os certificados pessoais assinados por esse certificado de autoridade de certificação. Considere cautelosamente em quais Autoridades de Certificação você quer confiar e inclua apenas o conjunto de certificados de CA necessários para autenticar seus clientes e gerenciadores. Não é recomendado incluir o conjunto completo de certificados de CA padrão, a menos que esse seja um requisito definitivo para sua política de segurança.

Localizando o Repositório de Chaves para um Gerenciador de Filas em Sistemas UNIX, Linux, and Windows

Use este procedimento para obter o local do arquivo do banco de dados de chave do gerenciador de filas

Procedimento

1. Exiba os atributos do seu gerenciador de filas, usando um dos seguintes comandos MQSC:

```
DISPLAY QMGR ALL
DISPLAY QMGR SSLKEYR
```

   Também é possível exibir os atributos do gerenciador de filas usando os comandos Explorer ou PCF do IBM WebSphere MQ.
2. Examine a saída do comando para o nome do caminho e de raiz do arquivo de banco de dados da chave.
   Por exemplo,
   a. nos sistemas UNIX and Linux: `/var/mqm/qmgrs/QM1/ssl/key`, em que `/var/mqm/qmgrs/QM1/ssl` é o caminho e `key` é o nome do stem
   b. no Windows: *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl\key`, em que *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl` é o caminho e `key` é o nome da raiz *MQ_INSTALLATION_PATH* Representa o diretório de alto nível no qual o WebSphere MQ está instalado

Mudando o local do repositório de chaves para um gerenciador de filas em sistemas UNIX, Linux ou Windows

É possível alterar o local do arquivo de banco de dados de chave do gerenciador de filas por vários meios, incluindo o comando MQSC ALTER QMGR.

É possível alterar o local do arquivo de banco de dados de chave do seu gerenciador de filas, ao usar o comando MQSC ALTER QMGR para configurar o atributo do repositório de chaves do seu gerenciador de filas. Por exemplo, em sistemas UNIX and Linux :

```
ALTER QMGR SSLKEYR('/var/mqm/qmgrs/QM1/ssl/MyKey')
```

O arquivo de banco de dados chave tem o nome completo do arquivo: `/var/mqm/qmgrs/QM1/ssl/MyKey.kdb`

No Windows:

```
ALTER QMGR SSLKEYR('C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\Mykey')
```

O arquivo de banco de dados chave tem o nome completo do arquivo: `C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\Mykey.kdb`

Atenção: Assegure-se de não incluir a extensão `.kdb` no nome do arquivo na palavra-chave SSLKEYR, já que o gerenciador de filas anexa esta extensão automaticamente.

Também é possível alterar os atributos de seu gerenciador de filas usando os comandos WebSphere MQ Explorer ou PCF.

Ao alterar o local de um arquivo de banco de dados de chave do gerenciador de filas, os certificados não serão transferidos a partir do local antigo. Se o arquivo do banco de dados de chaves que você está acessando agora for um novo arquivo do banco de dados de chave, deve-se preenchê-lo com os certificados de CA e pessoais necessários, conforme descrito em “Importando um certificado pessoal em um repositório de chaves em sistemas UNIX, Linux, and Windows” na página 136.

Localizando o Repositório de Chaves para um Cliente MQI do IBM WebSphere MQ em Sistemas UNIX, Linux, and Windows

O local do repositório de chaves é fornecido pela variável MQSSLKEYR ou especificado na chamada MQCONNX.

Examine a variável de ambiente MQSSLKEYR para obter o local de seu arquivo do banco de dados de chave do cliente MQI do IBM WebSphere MQ .. Por exemplo:

```
echo $MQSSLKEYR
```

Verifique também seu aplicativo, pois o nome do arquivo do banco de dados de chaves também pode ser configurado em uma chamada MQCONNX, conforme descrito em “Especificando o Local do Repositório de Chaves para um IBM WebSphere MQ Cliente MQI em Sistemas UNIX, Linux, and Windows” na página 123. O valor definido em uma chamada MQCONNX substitui o valor de MQSSLKEYR.

Especificando o Local do Repositório de Chaves para um IBM WebSphere MQ Cliente MQI em Sistemas UNIX, Linux, and Windows

Não há repositório de chaves padrão para um cliente MQI IBM WebSphere MQ . É possível especificar seu local em uma de duas maneiras. Certifique-se de que o arquivo do banco de dados de chaves somente possa ser acessado por usuários ou administradores pretendidos para evitar cópias não-autorizadas para outros sistemas.

É possível especificar o local do arquivo do banco de dados de chaves do cliente MQI do IBM WebSphere MQ de uma das duas maneiras:

- Configurar a variável de ambiente MQSSLKEYR. Por exemplo, em sistemas UNIX and Linux :

  ```
  export MQSSLKEYR=/var/mqm/ssl/key
  ```

  O arquivo de banco de dados de chave tem o nome completo de arquivo:

```
/var/mqm/ssl/key.kdb
```

No Windows:

```
set MQSSLKEYR=C:\Program Files\IBM\WebSphere MQ\ssl\key
```

O arquivo de banco de dados de chave tem o nome completo de arquivo:

```
C:\Program Files\IBM\WebSphere MQ\ssl\key.kdb
```

Nota: A extensão **.**kdb é uma parte obrigatória do nome do arquivo, mas não é incluída como parte do valor da variável de ambiente.

- Forneça o nome do caminho e de raiz do arquivo de banco de dados de chaves no campo *KeyRepository* da estrutura do MQSCO quando um aplicativo executar uma chamada MQCONNX. Para obter mais informações sobre como usar a estrutura MQSCO em MQCONNX, consulte Visão geral para MQSCO ..

Quando as mudanças nos certificados ou no armazenamento de certificados entram em vigor nos sistemas UNIX, Linux ou Windows.

Ao alterar os certificados em um armazenamento de certificados, ou o local do armazenamento de certificados, as mudanças entrarão em vigor, dependendo do tipo de canal e de como o canal está sendo executado.

Mudanças nos certificados no arquivo do banco de dados de chave e no atributo de repositório de chaves entrarão em vigor nas seguintes situações:

- Quando um novo processo de canal único de saída executar primeiramente um canal SSL.
- Quando um novo processo de canal único de TCP/IP de entrada receber primeiramente um pedido para iniciar um canal SSL.
- Quando o comando MQSC REFRESH SECURITY TYPE(SSL) é emitido para atualizar o ambiente SSL do Websphere MQ.
- Para processos do aplicativo cliente, quando a última conexão SSL no processo é fechada. A próxima conexão SSL selecionará as mudanças do certificado.
- Para canais que são executados como encadeamentos de processo pooling process (amqrmppa), quando o processo pooling process for iniciado ou reiniciado e executar primeiramente um canal SSL. Se o processo pooling process já executou um canal SSL e desejar que a alteração entre em vigor imediatamente, execute o comando MQSC REFRESH SECURITY TYPE(SSL).
- Para canais que são executados como encadeamentos do inicializador de canal, quando o inicializador de canal for iniciado ou reiniciado e executar primeiramente um canal SSL. Se o processo de inicializador de canal já executou um canal SSL e desejar que a alteração entre em vigor imediatamente, execute o comando MQSC REFRESH SECURITY TYPE(SSL).
- Para canais que são executados como encadeamentos de um listener TCP/IP, quando o listener for iniciado ou reiniciado e receber primeiramente um pedido para iniciar um canal SSL. Se o listener já executou um canal SSL e desejar que a alteração entre em vigor imediatamente, execute o comando MQSC REFRESH SECURITY TYPE(SSL).

Também é possível atualizar o ambiente SSL do WebSphere MQ usando os comandos IBM WebSphere MQ Explorer ou PCF.

Criando um certificado pessoal auto-assinado em sistemas UNIX, Linux, and Windows

É possível criar um certificado autoassinado usando iKeyman, iKeycmdou runmqakm.

Nota: O IBM WebSphere MQ não suporta algoritmos SHA-3 ou SHA-5. É possível usar os nomes de algoritmos de assinatura digital SHA384WithRSA e SHA512WithRSA porque ambos os algoritmos são membros da família do SHA-2.

Os nomes dos algoritmos de assinatura digital SHA3WithRSA e SHA5WithRSA são descontinuados porque eles são uma forma abreviada do SHA384WithRSA e do SHA512WithRSA, respectivamente.

Para obter mais informações sobre por que você pode desejar usar certificados autoassinados, consulte “Usando certificados autoassinados para autenticação mútua de dois gerenciadores de filas” na página 210.

Nem todos os certificados digitais podem ser usados com todos os CipherSpecs. Assegure-se de criar um certificado que seja compatível com os CipherSpecs que precisa usar. WebSphere MQ suporta três tipos diferentes de CipherSpec. Para obter detalhes, consulte “Interoperabilidade da curva elíptica e do RSA de CipherSpecs” na página 36 no tópico “Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ” na página 35. Para usar o CipherSpecs do Tipo 1 (aqueles com nomes que começam com ECDHE_ECDSA_), deve-se usar o comando **runmqakm** para criar o certificado e deve-se especificar um parâmetro de algoritmo de assinatura ECDSA da Curva Elíptica; por exemplo **-sig_alg** EC_ecdsa_with_SHA384 .

Usando o iKeyman

iKeyman não fornece uma opção compatível com FIPS. Se precisar gerenciar certificados SSL ou TLS de uma maneira que seja compatível com FIPS, use o comando **runmqakm**.

Use o procedimento a seguir para obter um certificado autoassinado para seu gerenciador de fila ou cliente MQI do WebSphere MQ :

1. Inicie a GUI do iKeyman usando o comando **strmqikm**.
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é exibida.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo de banco de dados de chave no qual você quer salvar o certificado, por exemplo key.kdb.
6. Click **Open**. A janela Prompt de Senha é exibida.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**.
8. No menu **Criar** clique em **Novo Certificado Auto-Assinado**. A janela Criar Novo Certificado Autoassinado é exibida.
9. No campo **Rótulo chave** , digite:
   - Para um gerenciador de fila, ibmwebspheremq seguido pelo nome de seu gerenciador de filas dobrado para minúsculas. Por exemplo, para QM1, ibmwebspheremqqm1ou,
   - Para um cliente do WebSphere MQ , ibmwebspheremq seguido por seu ID do usuário de logon dobrado para minúsculas, por exemplo ibmwebspheremqmyuserid .
10. Digite ou selecione um valor para qualquer campo no **Distinguished name**ou qualquer um dos campos **Subject alternative name** .
11. Para os campos restantes, aceite os valores padrão ou digite ou selecione novos valores. Para obter mais informações sobre nomes distintos, consulte “Nomes Distintos” na página 11.
12. Clique em **OK**. A lista **Certificados Pessoais** mostra o rótulo certificado pessoal auto-assinado que você criou.

Usando a linha de comandos

Use os comandos a seguir para criar um certificado pessoal autoassinado usando iKeycmd ou runmqakm:

- Usando iKeycmd em sistemas UNIX, Linux e Windows :

```
runmqckm -cert -create -db filename -pw
password -label label
```

```
        -dn distinguished_name -size key_size
 -x509version version -expire days
 -sig_alg algorithm
```

Em vez de -dn *distinguished_name* , é possível usar -san_dsname *DNS_names* , -san_emailaddr *email_addresses* ou -san_ipaddr *IP_addresses* .

- Usando runmqakm:

```
runmqakm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days

        -fips -sig_alg algorithm
```

-db *filename*	O nome completo do arquivo de um banco de dados de chave CMS.
-pw *password*	A senha para o banco de dados de chave CMS.
-label *label*	O rótulo de chave anexado ao certificado.
-dn *distinguished_name*	O nome distinto X.500 colocado entre aspas duplas. Pelo menos um atributo é necessário. É possível fornecer diversos atributos OU ou DC.
-size *key_size*	O tamanho da chave. Para iKeycmd, o valor pode ser 512 ou 1024. Para runmqakm, o valor pode ser 512, 1024, 2048 ou 4096.
-x509version *version*	A versão do certificado X.509 a ser criado. O valor pode ser de 1, 2 ou 3. O padrão é 3.
-expire *days*	O prazo de expiração em dias do certificado. O padrão é 365 dias para um certificado.
-fips	Especifica que o comando é executado no modo FIPS. Este modo desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando estiver no modo FIPS, o componente ICC usará os algoritmos que foram validados pelo FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.
-sig_alg	Para runmqakm, o algoritmo hash usado durante a criação de um certificado autoassinado. Este algoritmo hash é usado para criar a assinatura associada ao certificado auto-assinado recém criado. O valor pode ser 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. O valor padrão é SHA1WithRSA.
-sig_alg	Para iKeycmd, o algoritmo de assinatura assimétrica usado para a criação do par de chaves da entrada. O valor pode ser 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, A_WITH_RSA , SHAWithDSAou SHAWithRSA. O valor padrão é SHA1WithRSA.

`-san_dnsname` *DNS_names*	Uma lista delimitada por vírgula ou espaço com os nomes de DNS para a entrada sendo criada.
`-san_emailaddr` *email_addresses*	Uma lista delimitada por vírgula ou espaço com os endereços de e-mail para a entrada sendo criada.
`-san_ipaddr` *IP_addresses*	Uma lista delimitada por vírgula ou espaço com os endereços IP para a entrada sendo criada.

distributed

Solicitando um certificado pessoal nos sistemas UNIX, Linux, and Windows

É possível solicitar um certificado pessoal usando o **strmqikm** (iKeyman) GUI ou na linha de comandos usando os comandos **runmqckm** ou **runmqakm** . Se precisar gerenciar certificados SSL ou TLS de uma maneira que seja compatível com FIPS, use o comando **runmqakm**.

Sobre esta tarefa

É possível solicitar um certificado pessoal usando a GUI do iKeyman ou pela linha de comandos, sujeito às seguintes considerações:

- O WebSphere MQ não suporta algoritmos SHA-3 ou SHA-5. É possível usar os nomes de algoritmos de assinatura digital SHA384WithRSA e SHA512WithRSA porque ambos os algoritmos são membros da família do SHA-2.
- Os nomes dos algoritmos de assinatura digital SHA3WithRSA e SHA5WithRSA são descontinuados porque eles são uma forma abreviada do SHA384WithRSA e do SHA512WithRSA, respectivamente.
- Nem todos os certificados digitais podem ser usados com todos os CipherSpecs. Assegure-se de solicitar um certificado que seja compatível com os CipherSpecs que precisa usar. O WebSphere MQ suporta três tipos diferentes de CipherSpec. Para obter detalhes, consulte "Interoperabilidade da curva elíptica e do RSA de CipherSpecs" na página 36 no tópico "Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ" na página 35.
- Para usar o Tipo 1 CipherSpecs (com nomes começando com ECDHE_ECDSA_), deve-se usar o comando **runmqakm** para solicitar o certificado e deve-se especificar um parâmetro de algoritmo de assinatura ECDSA de Curva Elíptica; por exemplo, **-sig_alg** EC_ecdsa_with_SHA384.
- Somente o comando runmqakm fornece uma opção compatível com FIPS.
- Se você estiver usando o hardware de criptografia, consulte "Solicitando um Certificado Pessoal para o Hardware PKCS #11" na página 143.

Usando a interface com o usuário do iKeyman

Sobre esta tarefa

iKeyman não fornece uma opção compatível com FIPS. Se precisar gerenciar certificados SSL ou TLS de uma maneira que seja compatível com FIPS, use o comando **runmqakm**.

Procedimento

Conclua as etapas a seguir para se candidatar a um certificado pessoal usando a interface com o usuário iKeyman:

1. Inicie a interface com o usuário do iKeyman usando o comando **strmqikm**
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**.
   A janela **Abrir** é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo do banco de dados de chaves a partir do qual você quer gerar seu pedido; por exemplo, key.kdb.
6. Click **Open**.

A janela **Prompt de senha** é aberta.

7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**.

   O nome do seu arquivo do banco de dados de chave é mostrado no campo **Nome do arquivo**.
8. No menu **Criar** clique em **Novo Pedido de Certificado**. A janela **Criar nova chave e solicitação de certificado** é aberta.
9. No campo **Rótulo chave** , insira os rótulos a seguir:
   - Para um gerenciador de filas, insira `ibmwebspheremq` seguido pelo nome de seu gerenciador de filas alterado para minúsculas. Por exemplo, para um gerenciador de filas chamado `QM1`, insira `ibmwebspheremqqm1`
   - Para um IBM WebSphere MQ MQI client, insira `ibmwebspheremq` seguido por seu ID do usuário de logon, tudo em minúsculas; por exemplo, `ibmwebspheremqmyuserid` .
10. Digite ou selecione um valor para qualquer campo no campo **Nome distinto** ou qualquer um dos campos **Nome alternativo do assunto**. Para os campos restantes, seja aceito os valores padrão ou insira ou selecione novos valores.

    Para obter mais informações sobre Nomes Distintos, consulte “Nomes Distintos” na página 11.
11. No campo **Inserir o nome de um arquivo no qual armazenar a solicitação de certificado**, aceite o padrão `certreq.arm` ou insira um novo valor com um caminho completo.
12. Clique em **OK**.

    Uma janela de confirmação é exibida.
13. Clique em **OK**.

    A lista **Pedidos de Certificado Pessoal** mostra o rótulo do novo pedido de certificado pessoal que você criou. O pedido de certificado é armazenado no arquivo que você escolheu na etapa “11” na página 128.
14. Solicite o novo certificado pessoal enviando o arquivo para uma autoridade de certificação (CA) ou copiando o arquivo no formulário de solicitação no website para a CA.

Usando a linha de comandos

Procedimento

Use os comandos a seguir para solicitar um certificado pessoal usando o comando **runmqckm** ou o comando **runmqakm**:

- Usando o **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -sig_alg algorithm
```

  Em vez de `-dn` *distinguished_name* , é possível usar `-san_dsname` *DNS_names* , `-san_emailaddr` *email_addresses* ou `-san_ipaddr` *IP_addresses* .

- Usando runmqakm:

```
runmqakm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -fips
        -sig_alg algorithm
```

em que:

-db *filename*
Especifica o nome completo do arquivo de um banco de dados de chaves do CMS.

-pw *password*
Especifica a senha para o banco de dados de chaves do CMS.

-label *label*
Especifica o rótulo chave anexado ao certificado.

-dn *distinguished_name*
Especifica o nome distinto X.500 colocado entre aspas duplas. Pelo menos um atributo é necessário. É possível fornecer diversos atributos OU e DC.

-size *key_size*
Especifica o tamanho da chave. Se você estiver usando **runmqckm** , o valor poderá ser 512 ou 1024. Se estiver usando **runmqakm**, o valor pode ser 512, 1024 ou 2048.

-file *filename*
Especifica o nome do arquivo para a solicitação de certificado.

-fips
Especifica que o comando é executado no modo FIPS. Este modo desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando em modo FIPS, o componente ICC usa algoritmos que são validados para FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

-sig_alg
Para **runmqckm**, especifica o algoritmo de assinatura assimétrica usado para a criação do par de chaves da entrada. The value can be 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, or SHAWithRSA . O valor padrão é SHA1WithRSA

-sig_alg
Para o **runmqakm**, especifica o algoritmo hash usado durante a criação de uma solicitação de certificado. Este algoritmo hash é usado para criar a assinatura associada a solicitação de certificado criada recentemente. O valor pode ser 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. O valor padrão é SHA1WithRSA.

-san_dnsname *DNS_names*
Especifica uma lista de nomes DNS delimitada por vírgulas ou espaços para a entrada sendo criada.

-san_emailaddr *email_addresses*
Especifica uma lista de endereços de e-mail delimitada por vírgulas ou espaços para a entrada sendo criada.

-san_ipaddr *IP_addresses*
Especifica uma lista de endereços IP delimitada por vírgulas ou espaços para a entrada sendo criada.

Renovando um certificado pessoal existente em sistemas UNIX, Linux, and Windows

É possível renovar um certificado pessoal usando a interface com o usuário iKeyman ou usando os comandos **iKeycmd** ou **runmqakm** .

Antes de começar

Se você tiver um requisito para usar tamanhos de chaves maiores para seus certificados pessoais, as etapas de renovação descritas abaixo não funcionarão, porque a solicitação de certificado recriada é gerada a partir de uma chave existente

Siga as etapas descritas em "Solicitando um certificado pessoal nos sistemas UNIX, Linux, and Windows" na página 127 para criar uma nova solicitação de certificado, usando os tamanhos de chave necessários. Esse processo substitui a sua chave existente

Sobre esta tarefa

Um certificado pessoal possui uma data de expiração, após a qual o certificado não pode mais ser usado. Esta tarefa explica como renovar um certificado pessoal existente antes de ele expirar.

Usando a interface com o usuário do iKeyman

Sobre esta tarefa

iKeyman não fornece uma opção compatível com FIPS. Se precisar gerenciar certificados SSL ou TLS de uma maneira que seja compatível com FIPS, use o comando **runmqakm**.

Procedimento

Conclua as etapas a seguir para se candidatar a um certificado pessoal usando a interface com o usuário iKeyman:

1. Inicie a interface com o usuário do iKeyman usando o comando **strmqikm** em sistemas UNIX, Linux, and Windows
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**.

   A janela **Abrir** é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo do banco de dados de chaves a partir do qual você quer gerar seu pedido; por exemplo, `key.kdb`.
6. Click **Open**.

   A janela **Prompt de senha** é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**.

   O nome do seu arquivo do banco de dados de chave é mostrado no campo **Nome do arquivo**.
8. Selecione **Certificados pessoais** a partir do menu suspenso de seleção e selecione o certificado que você deseja renovar a partir da lista.
9. Clique em **Recrear solicitação ...** .

   Uma janela é aberta para que você insira o nome do arquivo e as informações de local do arquivo.
10. No campo **file name**, aceite o padrão `certreq.arm`ou digite um novo valor, incluindo o caminho de arquivo completo.
11. Clique em **OK**. A solicitação de certificado é armazenada no arquivo selecionado na etapa “9” na página 130.
12. Solicite o novo certificado pessoal enviando o arquivo para uma autoridade de certificação (CA) ou copiando o arquivo no formulário de solicitação no website para a CA.

Usando a linha de comandos

Procedimento

Use os comandos a seguir para solicitar um certificado pessoal usando os comandos **iKeycmd** ou o **runmqakm**:

- Usando o **iKeycmd** em sistemas UNIX, Linux, and Windows:

```
runmqckm -certreq -recreate -db filename -pw
password -label label
 -target filename
```

- Usando runmqakm:

```
runmqakm -certreq -recreate -db filename -pw
password -label label
  -target filename
```

em que:

-db *filename*
Especifica o nome completo do arquivo de um banco de dados de chaves do CMS.

-pw *password*
Especifica a senha para o banco de dados de chaves do CMS.

-target *filename*
Especifica o nome do arquivo para a solicitação de certificado.

Como proceder a seguir

Depois de ter recebido o certificado pessoal assinado da autoridade de certificação, será possível incluí-lo em seu banco de dados de chaves usando as etapas descritas em “Recebendo Certificados Pessoais em um Repositório de Chaves nos Sistemas UNIX, Linux e Windows” na página 131.

Recebendo Certificados Pessoais em um Repositório de Chaves nos Sistemas UNIX, Linux e Windows

Use este procedimento para receber um certificado pessoal no arquivo de banco de dados de chave. O repositório de chaves deve ser o mesmo repositório em que foi criada a solicitação de certificado.

Depois que a CA enviar um novo certificado pessoal, inclua-o ao arquivo de banco de dados de chave a partir do qual o novo pedido de certificado foi gerado. Se a CA enviar o certificado como parte de uma mensagem de e-mail, copie o certificado em um arquivo separado.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Certifique-se de que o arquivo de certificado a ser importado tenha permissão de gravação para o usuário atual e, em seguida, use o procedimento a seguir para um gerenciador de filas ou um cliente MQI do WebSphere MQ para receber um certificado pessoal no arquivo do banco de dados de chave:

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (no Windows UNIX and Linux).
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo de banco de dados de chave no qual você quer incluir o certificado, por exemplo key.kdb.
6. Clique em **Abrir** e, em seguida, clique em **OK**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**. Selecione a visualização **Certificados Pessoais**.
8. Clique em **Receber**. A janela Receber Certificado de um Arquivo é aberta.
9. Digite o nome do arquivo do certificado para o novo certificado pessoal ou clique em **Procurar** para selecionar o nome e a localização.
10. Clique em **OK**. Se você já tiver um certificado pessoal no banco de dados de chave, uma janela será aberta perguntando se você deseja configurar a chave que está incluindo como a chave padrão no banco de dados.
11. Clique em **Sim** ou **Não**. A janela Inserir um Rótulo é aberta.

12. Clique em **OK**. O campo**Certificados Pessoais** mostra o rótulo do novo certificado pessoal que você incluiu.

Usando a linha de comandos

Use os comandos a seguir para incluir um certificado pessoal em um arquivo de banco de dados de chaves usando iKeycmd :

- Em UNIX, Linux e Windows, emita o comando a seguir:

```
runmqckm -cert -receive -file filename -db filename -pw
password
        -format ascii
```

em que:

`-file filename`	é o nome qualificado do arquivo que contém o certificado pessoal.
`-db filename`	é o nome qualificado do arquivo de um banco de dados de chave CMS.
`-pw password`	é a senha do banco de dados de chave CMS.
`-format ascii`	é o formato do certificado. O valor pode ser `ascii` para Base64-encoded ASCII ou `binary` para dados DER binários. O padrão é `ascii` ..

Se você estiver usando o hardware de criptografia, consulte o "Importando um certificado pessoal para o hardware PKCS #11" na página 146.

Extraindo um Certificado de CA de um Repositório de Chaves

Siga este procedimento para extrair um certificado de CA.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Execute as seguintes etapas na máquina a partir da qual deseja extrair o certificado de CA:

1. Inicie a GUI do iKeyman usando o comando **strmqikm**.
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo do banco de dados de chave a partir do qual você deseja extrair, por exemplo `key.kdb`.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**.
8. No campo **Conteúdo do banco de dados de chave**, selecione **Certificados do Signatário** e selecione o certificado que deseja extrair.
9. Clique em **Extrair**. A janela Extrair um Certificado para um Arquivo é aberta.
10. Selecione o **Tipo de Dados** do certificado, por exemplo **Dados ASCII codificados na Base64** para um arquivo com a extensão `.arm`.
11. Digite o nome do arquivo do certificado e a localização onde você quer armazenar o certificado, ou clique em **Procurar** para selecionar o nome e a localização.
12. Clique em **OK**. O certificado é gravado no arquivo que você especificou.

Usando a linha de comandos

Use os comandos a seguir para extrair um certificado de autoridade de certificação usando iKeycmd :

- No UNIX, Linux e Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

em que:

-db *filename*	é o nome qualificado do caminho de um banco de dados de chave CMS.
-pw *password*	é a senha do banco de dados de chave CMS.
-label *label*	é o rótulo anexado ao certificado.
-target *filename*	é o nome do arquivo de destino.
-format *ascii*	é o formato do certificado. O valor pode ser `ascii` para ASCII codificado na Base64 ou `binary` para dados DER binários. O padrão é `ascii`.

Extraindo a parte pública de um certificado autoassinado de um repositório de chaves nos sistemas UNIX, Linux e Windows

Siga este procedimento para extrair a parte pública de um certificado autoassinado.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Execute as seguintes etapas na máquina a partir da qual você deseja extrair a parte pública de um certificado autoassinado:

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (em UNIX Linux e Windows).
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo de banco de dados da chave a partir do qual você deseja extrair o certificado, por exemplo, `key.kdb`.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**.
8. No campo **Conteúdo do Banco de Dados de Chaves**, selecione **Certificados Pessoais** e selecione o certificado.
9. Clique **Extrair Certificado**. A janela Extrair um Certificado para um Arquivo é aberta.
10. Selecione o **Tipo de Dados** do certificado, por exemplo **Dados ASCII codificados na Base64** para um arquivo com a extensão `.arm`.
11. Digite o nome do arquivo do certificado e a localização onde você quer armazenar o certificado, ou clique em **Procurar** para selecionar o nome e a localização.
12. Clique em **OK**. O certificado é gravado no arquivo que você especificou. Observe que ao extrair um certificado (em vez de exportar), apenas a parte pública do certificado é incluída, de modo que uma senha não seja necessária.

Usando a linha de comandos

Use os comandos a seguir para extrair a parte pública de um certificado autoassinado usando iKeycmd ou runmqakm:

- No UNIX, Linux e Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

- Usando runmqakm:

```
runmqakm -cert -extract -db filename -pw password -label label
            -target filename -format ascii -fips
```

em que:

-db *filename*	é o nome qualificado do caminho de um banco de dados de chave CMS.
-pw *password*	é a senha do banco de dados de chave CMS.
-label *label*	é o rótulo anexado ao certificado.
-target *filename*	é o nome do arquivo de destino.
-format *ascii*	é o formato do certificado. O valor pode ser `ascii` para ASCII codificado na Base64 ou `binary` para dados DER binários. O padrão é `ascii`.

Incluindo um certificado de autoridade de certificação (ou a parte pública de um certificado autoassinado) em um repositório de chaves, em sistemas UNIX, Linux, and Windows

Siga este procedimento para incluir um certificado de CA ou a parte pública de um certificado autoassinado no repositório de chaves.

Se o certificado que deseja incluir estiver em uma cadeia de certificados, será necessário incluir também todos os certificados que estão acima dele na cadeia. É necessário incluir os certificados em ordem estritamente decrescente iniciando da raiz, seguido pelo certificado de CA logo abaixo dela na cadeia, e assim por diante.

Onde as instruções a seguir se referirem a um certificado de CA, elas também se aplicarão à parte pública de um certificado autoassinado.

Nota: Se o certificado que deseja incluir estiver em uma cadeia de certificados, você deverá também incluir todos os certificados que estão acima dele na cadeia. Deve-se garantir que o certificado esteja em codificação ASCII (UTF-8) ou binária (DER), pois o IBM Global Secure Toolkit (GSKit) não suporta certificados com outros tipos de codificação. Você deve incluir os certificados em ordem estritamente decrescente iniciando da raiz, seguido pelo certificado CA imediatamente abaixo dele na cadeia.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Execute as seguintes etapas na máquina na qual deseja incluir o certificado de CA:

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (nos sistemas UNIX Linux e Windows).
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.

5. Selecione o arquivo de banco de dados de chave no qual você quer incluir o certificado, por exemplo key.kdb.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do arquivo de banco de dados de chave é exibido no campo **Nome de Arquivo**.
8. No campo **Conteúdo do banco de dados de chave**, selecione **Certificados de Assinante**.
9. Clique em **Incluir**. A janela Incluir Certificado de CA de um Arquivo é aberta.
10. Digite o nome do arquivo do certificado e a localização em que está armazenado, ou clique em **Procurar** para selecionar o nome e a localização.
11. Clique em **OK**. A janela Inserir um Rótulo é aberta.
12. Na janela Digite um Rótulo, digite o nome do certificado.
13. Clique em **OK**. O certificado é incluído ao banco de dados de chave.

Usando a linha de comandos

Use os seguintes comandos para incluir um certificado de CA usando iKeycmd :

- Em UNIX, Linux e Windows, emita o comando a seguir:

```
runmqckm -cert -add -db filename -pw password -label label -file filename
        -format ascii
```

em que:

-db *filename*	é o nome do caminho completo do banco de dados de chave CMS.
-pw *password*	é a senha do banco de dados de chave CMS.
-label *label*	é o rótulo anexado ao certificado.
-file *filename*	é o nome do arquivo que contém o certificado.
-format *ascii*	é o formato do certificado. O valor pode ser ascii para ASCII codificado na Base64 ou binary para dados DER binários. O padrão é ascii.

Exportando um Certificado Pessoal de um Repositório de Chaves

Siga este procedimento para exportar um certificado pessoal.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Execute as seguintes etapas na máquina a partir da qual você deseja exportar o certificado pessoal:

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (no Windows UNIX and Linux .
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo do banco de dados de chaves a partir do qual você deseja exportar o certificado, por exemplo key.kdb.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**.

8. No campo **Conteúdo do banco de dados de chave**, selecione **Certificados Pessoais** e selecione o certificado que deseja exportar.
9. Clique em **Exportar/Importar**. A janela Exportar/Importar chave é aberta.
10. Selecione **Exportar Chave**.
11. Selecione o **Tipo de arquivo de chave** do certificado que deseja exportar, por exemplo **PKCS12**.
12. Digite o nome do arquivo e o local para o qual deseja exportar o certificado, ou clique em **Procurar** para selecionar o nome e o local.
13. Clique em **OK**. A janela Prompt de Senha é aberta. Observe que quando o certificado é exportado (em vez de extraído), ambas as partes pública e privada do certificado são incluídas. Eis o motivo pelo qual o arquivo exportado é protegido por uma senha. Ao extrair um certificado, apenas a parte pública do certificado é incluída, de modo que uma senha não seja necessária.
14. Digite a senha no campo **Senha** e digite-a novamente no campo **Confirmar Senha**.
15. Clique em **OK**. O certificado é exportado para o arquivo que você especificou.

Usando a linha de comandos

Use os seguintes comandos para exportar um certificado pessoal usando iKeycmd:

- No UNIX, Linux e Windows:

```
runmqckm -cert -export -db filename -pw password -label label -type cms
        -target filename -target_pw password -target_type pkcs12
```

em que:

`-db` *`filename`*	é o nome do caminho completo do banco de dados de chave CMS.
`-pw` *`password`*	é a senha do banco de dados de chave CMS.
`-label` *`label`*	é o rótulo anexado ao certificado.
`-type` *`cms`*	é o tipo do banco de dados.
`-target` *`filename`*	é o nome do caminho completo do arquivo de destino.
`-target_pw` *`password`*	é a senha para criptografar o certificado.
`-target_type` *`pkcs12`*	é o tipo do certificado.

Importando um certificado pessoal em um repositório de chaves em sistemas UNIX, Linux, and Windows

Siga este procedimento para importar um certificado pessoal

Antes de importar um certificado pessoal no formato PKCS #12 para o arquivo do banco de dados de chave, deve-se primeiramente incluir a cadeia válida completa da emissão de certificados de CA para o arquivo de banco de dados chave (consulte “Incluindo um certificado de autoridade de certificação (ou a parte pública de um certificado autoassinado) em um repositório de chaves, em sistemas UNIX, Linux, and Windows” na página 134).

Os arquivos PKCS #12 devem ser considerados temporariamente e excluídos após o uso.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

Execute as seguintes etapas na máquina para a qual deseja importar o certificado pessoal:

1. Inicie a GUI do iKeyman usando o comando **strmqikm**.
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é exibida.

3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo de banco de dados de chave no qual você quer incluir o certificado, por exemplo `key.kdb`.
6. Click **Open**. A janela Prompt de Senha é exibida.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do arquivo de banco de dados de chave é exibido no campo **Nome de Arquivo**.
8. No campo **Conteúdo do banco de dados de chave**, selecione **Certificados Pessoais**.
9. Se houver certificados na visualização Certificados Pessoais, siga estas etapas:
   a. Clique em **Exportar/Importar**. A janela Exportar/Importar chave é exibida.
   b. Selecione **Importar Chave**.
10. Se não houver certificados na visualização Certificados Pessoais, clique em **Importar**.
11. Selecione **Tipo de Arquivo-chave** do certificado que deseja importar, por exemplo, PKCS12.
12. Digite o nome do arquivo do certificado e a localização em que está armazenado, ou clique em **Procurar** para selecionar o nome e a localização.
13. Clique em **OK**. A janela Prompt de Senha é exibida.
14. No campo **Senha**, digite a senha usada quando o certificado foi exportado.
15. Clique em **OK**. A janela Alterar Rótulos é exibida. Esta janela permite que os rótulos dos certificados que estão sendo importados sejam alterados se, por exemplo, um certificado com o mesmo rótulo já existir no banco de dados de chaves de destino. A alteração dos rótulos de certificado não possui efeito na validação da cadeia de certificados. Isso pode ser usado para alterar o rótulo do certificado pessoal para aquele requerido pelo WebSphere MQ para associar o certificado ao gerenciador de filas ou cliente específico (`ibmwebspheremqqm1` por exemplo).
16. Para alterar um rótulo, selecione o rótulo necessário da lista **Selecionar um rótulo para alterar**. O rótulo é copiado no campo de entrada **Inserir um novo rótulo**. Substitua o texto do rótulo pelo texto do novo rótulo e clique em **Aplicar**.
17. O texto no campo de entrada **Inserir um novo rótulo** é copiado de volta no campo **Selecionar um rótulo para alterar**, substituindo o rótulo selecionado originalmente e, assim, designando um novo rótulo ao certificado correspondente.
18. Ao alterar todos os rótulo necessários, clique em **OK**. A janela Alterar Rótulos é fechada e a janela original do IBM Key Management reaparece com os campos **Certificados Pessoais** e **Certificados Signatários** atualizados com os certificados corretamente rotulados.
19. O certificado é importado para o banco de dados de chave de destino.

Usando a linha de comandos

Para importar um certificado pessoal usando iKeycmd, use os seguintes comandos:

- No UNIX, Linux e Windows:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label
```

em que:

`-file` *filename*	é o nome qualificado do arquivo que contém o certificado PKCS #12.
`-pw` *password*	é a senha do certificado PKCS #12.
`-type` *pkcs12*	é o tipo do arquivo.
`-target` *filename*	é o nome do banco de dados de chave CMS de destino.

`-target_pw` *password*	é a senha do banco de dados de chave CMS.
`-target_type` *cms*	é o tipo de banco de dados especificado por `-target`
`-label` *label*	é o rótulo certificado que será importado a partir do banco de dados de chave de origem.
`-new_label` *label*	é o rótulo que será atribuído ao certificado no banco de dados de destino. Se a opção `-new_label` for omitida, o padrão será usar a mesma opção `-label`.

O iKeycmd não fornece um comando para alterar rótulos de certificado diretamente. Use as seguintes etapas para alterar um rótulo de certificado:

1. Exporte o certificado para um arquivo PKCS #12 usando o comando **`-cert -export`** Especifique o rótulo certificado existente para a opção `-label`.
2. Remova a cópia existente do certificado do banco e dados de chave original usando o comando **`-cert -delete`**.
3. Importe o certificado do arquivo PKCS #12 usando o comando **`-cert -import`** . Especifique o rótulo antigo para a opção `-label` e o novo rótulo necessário para a opção `-new_label`. O certificado será importado de volta para o banco de dados de chave com o rótulo necessário.

Importando de um arquivo Microsoft .pfx

Siga este procedimento para transportar a partir de um arquivo Microsoft .pfx usando iKeyman. Não é possível utilizar runmqakm para importar um arquivo .pfx.

Um arquivo .pfx pode conter dois certificados relacionados à mesma chave. Um certificado é um certificado pessoal ou de site (contendo ambas as chaves pública e privada). O outro é o certificado (signatário) de CA (contendo apenas uma chave pública). Esses certificados não podem coexistir no mesmo arquivo de banco de dados de chave CMS, portanto, apenas um deles poderá ser importado. Além disso, o "nome fácil" ou rótulo é anexado somente ao certificado de assinante.

O certificado pessoal é identificado por um UUID (Unique User Identifier) gerado pelo sistema. Esta seção mostra a importação de um certificado pessoal de um arquivo pfx ao rotulá-lo com o nome amigável anteriormente atribuído ao certificado (signatário) de CA. Os certificados (signatário) de CA já deverão ter sido incluídos no banco de dados de chave de destino. Observe que os arquivos PKCS#12 devem ser considerados temporariamente e excluídos depois do uso.

Siga essas etapas para importar um certificado pessoal de um banco de dados de chave pfx de origem:

1. Inicie a GUI do iKeyman usando o comando **`strmqikm`** (em Linux, UNIX ou Windows). A janela IBM Key Management é exibida.
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é exibida.
3. Selecione um tipo de banco de dados de chave de **PKCS12**.
4. **É recomendável fazer um backup do banco de dados pfx antes de executar esta etapa.** Selecione o banco de dados de chave pfx que deseja importar. Clique em **Abrir**. A janela Prompt de Senha é exibida.
5. Digite a senha do banco de dados de chave e clique em **OK**. A janela IBM Key Management é exibida. A barra de títulos mostra o nome do arquivo do banco de dados de chave pfx selecionado, indicando que o arquivo está aberto e pronto para uso.
6. Selecione **Certificados do Signatário** na lista. O "nome fácil" do certificado necessário é exibido como um rótulo no painel Certificado de assinante.
7. Selecione a entrada do rótulo e clique em **Excluir** para remover o certificado do signatário. A janela de Confirmação é exibida.
8. Clique em **Sim**. O rótulo selecionado não é mais exibido no painel de Certificados do Signatário.
9. Repita as etapas 6, 7 e 8 para todos os certificados do signatário.
10. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é exibida.

11. Selecione o banco de dados CMS de chave de destino para o qual o arquivo pfx está sendo importado. Clique em **Abrir**. A janela Prompt de Senha é exibida.
12. Digite a senha do banco de dados de chave e clique em **OK**. A janela IBM Key Management é exibida. A barra de título mostra o nome do arquivo de banco de dados de chave selecionado, indicando que o arquivo está aberto e pronto.
13. Selecione **Certificados Pessoais** na lista.
14. Se houver certificados na visualização Certificados Pessoais, siga estas etapas:
    a. Clique em **Exportar/Importar chave**. A janela Exportar/Importar chave é exibida.
    b. Selecione **Importar** a partir Escolher tipo de ação.
15. Se não houver certificados na visualização Certificados Pessoais, clique em **Importar**.
16. Selecione o arquivo PKCS12.
17. Insira o nome do arquivo pfx como usado na Etapa 4. Clique em **OK**. A janela Prompt de Senha é exibida.
18. Especifique a mesma senha especificada quando o certificado de signatário foi excluído. Clique em **OK**.
19. A janela Alterar Rótulos é exibida (devendo ter apenas um único certificado disponível para importar). O rótulo certificado deve ser um UUID com o formato `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
20. Para alterar o rótulo, selecione o UUID no painel **Selecionar um rótulo para alterar:**. O rótulo será replicado no campo **Inserir novo rótulo:**. Substitua o texto do rótulo pelo texto do nome amigável que foi excluído na Etapa 7 e clique em **Aplicar**. O nome fácil deve estar no formato `ibmwebspheremq`, seguido do nome do gerenciador de filas ou do ID de logon do usuário do cliente MQI do WebSphere MQ em minúsculas.
21. Clique em **OK**. A janela Mudar Rótulos agora é removida e a janela original do IBM Key Management reaparece com os painéis Certificados Pessoais e Certificados Signatários atualizados com o certificado pessoal corretamente rotulado.
22. O certificado pessoal pfx será agora importado para o banco de dados de destino.

Não é possível mudar um rótulo certificado usando iKeycmd

Importando de um Arquivo PKCS #7

As ferramentas iKeyman e iKeycmd não suportam arquivos PKCS #7 (.p7b). Use a ferramenta runmqckm para importar certificados de um arquivo PKCS #7.

Use o seguinte comando para incluir o certificado de CA a partir de um arquivo PKCS #7:

```
runmqckm -cert -add -db filename -pw password -type cms -file filename
-label label
```

`-db` *`filename`*	é o nome qualificado do arquivo do banco de dados de chave CMS.
`-pw` *`password`*	é a senha do banco de dados de chave.
`-type` *`cms`*	é o tipo do banco de dados de chave.
`-file` *`filename`*	é o nome do arquivo PKCS #7.
`-label` *`label`*	é o rótulo atribuído ao certificado no banco de dados de destino. O primeiro certificado recebe o rótulo fornecido. Todos os outros certificados, se estiverem presentes, serão identificados com o nome do assunto.

Use o seguinte comando para importar um certificado pessoal a partir de um arquivo 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*	é o nome qualificado do arquivo que contém o certificado PKCS #7.
`-pw` *password*	é a senha do certificado PKCS #7.
`-type` *pkcs7*	é o tipo do arquivo.
`-target` *filename*	é o nome do banco de dados de chave de destino.
`-target_pw` *password*	é a senha do banco de dados de chave de destino.
`-target_type` *cms*	é o tipo de banco de dados especificado por `-target`
`-label` *label*	é o rótulo certificado que deve ser importado.
`-new_label` *label*	é o rótulo que será atribuído ao certificado no banco de dados de destino. Se a opção `-new_label` for omitida, o padrão é usar a mesma opção `-label`.

Excluindo um certificado de um repositório de chaves em sistemas UNIX, Linux, and Windows

Use este procedimento para remover certificados pessoais ou de CA.

Usando o iKeyman

Se for preciso gerenciar certificados SSL de um modo que seja compatível com FIPS, use o comando runmqakm. iKeyman não fornece uma opção compatível com FIPS.

1. Inicie a GUI do iKeyman usando o comando **strmqikm** (nos sistemas UNIX Linux e Windows).
2. A partir do menu **Arquivo de Banco de Dados de Chave**, clique em **Abrir**. A janela Abrir é aberta.
3. Clique em **Tipo de banco de dados de chave** e selecione **CMS** (Certificate Management System).
4. Clique em **Procurar** para navegar até o diretório que contém os arquivos de banco de dados da chave.
5. Selecione o arquivo do banco de dados de chaves a partir do qual você deseja excluir o certificado, por exemplo `key.kdb`.
6. Clique em **Abrir**. A janela Prompt de Senha é aberta.
7. Digite a senha definida ao criar o banco de dados de chave e clique em **OK**. O nome do seu arquivo de banco de dados de chave é exibido no campo **Nome do Arquivo**.
8. Na lista suspensa, selecione **Certificados pessoais** ou **Certificados de assinante**
9. Selecione o certificado que deseja excluir.
10. Se você ainda não possuir uma cópia do certificado e deseja salvá-lo, clique em **Exportar/Importar** e exporte-o (consulte “Exportando um Certificado Pessoal de um Repositório de Chaves” na página 135).
11. Com o certificado selecionado, clique em **Excluir**. A janela Confirmar é aberta.
12. Clique em **Sim**. O campo **Certificados Pessoais** não mostrará mais o rótulo certificado que você excluiu.

Usando a linha de comandos

Use os comandos a seguir para excluir um certificado usando o iKeycmd ou runmqakm:

- No UNIX, Linux e Windows:

```
runmqckm -cert -delete -db filename -pw password -label label
```

em que:

`-db` *filename*	é o nome qualificado do arquivo de um banco de dados de chave CMS.

-pw *password*	é a senha do banco de dados de chave CMS.
-label *label*	é o rótulo anexado ao certificado pessoal.
-fips	Especifica que o comando é executado no modo FIPS. Este modo desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando estiver no modo FIPS, o componente ICC usará os algoritmos que foram validados pelo FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

Gerando senhas fortes para proteção do repositório de chaves

É possível gerar senhas fortes para a proteção do repositório de chaves usando o comando **runmqakm**

É possível usar o comando **runmqakm** com os parâmetros a seguir para gerar uma senha forte:

```
runmqakm -random -create -length 14 -strong -fips
```

Ao usar a senha gerada no parâmetro **-pw** dos comandos de administração de certificados subsequentes, sempre coloque aspas duplas ao redor da senha. Nos sistemas UNIX and Linux, também deve-se usar um caractere de barra invertida para escapar os caracteres a seguir se eles aparecerem na sequência de senha:

```
!   \   "    '
```

Ao digitar a senha em resposta a um prompt a partir de **runmqckm**, **runmqakm** ou da GUI do iKeyman, então não é necessário citar ou escapar a senha. Não é necessário porque o shell do sistema operacional não afeta a entrada de dados nesses casos.

Configurando para hardware criptográfico em sistemas UNIX, Linux, and Windows

É possível configurar o hardware de criptografia para um gerenciador de filas ou cliente de várias maneiras.

É possível configurar o hardware de criptografia para um gerenciador de filas nos sistemas UNIX, Linux ou Windows usando um dos métodos a seguir:

- Use o comando ALTER QMGR MQSC com o parâmetro SSLCRYP, conforme descrito em ALTER QMGR.
- Use o IBM WebSphere MQ Explorer para configurar o hardware de criptografia no sistema UNIX, Linux ou Windows . Para obter mais informações, consulte a ajuda online.

É possível configurar o hardware de criptografia para um cliente WebSphere MQ em sistemas UNIX, Linux ou Windows usando um dos métodos a seguir:

- Configure a variável de ambiente MQSSLCRYP. Os valores permitidos para MQSSLCRYP são os mesmos do parâmetro SSLCRYP, conforme descrito em ALTER QMGR. Se você usar a versão GSK_PCS11 do parâmetro SSLCRYP, o rótulo do token PKCS #11 deverá ser especificado inteiramente em minúsculas.
- Configure o campo **CryptoHardware** da estrutura de opções de configuração de SSL, MQSCO, em uma chamada MQCONNX. Para obter mais informações, consulte Visão geral para MQSCO.

Se você configurou um hardware criptográfico que usa a interface PKCS #11 usando qualquer um desses métodos, será necessário armazenar o certificado pessoal para ser usado nos canais no arquivo de banco de dados de chave para o token de criptografia que você configurou. Isso é descrito no “Gerenciando certificados em hardware PKCS #11” na página 141.

Gerenciando certificados em hardware PKCS #11

É possível gerenciar certificados digitais no hardware de criptografia que suporta a interface PKCS #11.

Sobre esta tarefa

Deve-se criar um banco de dados de chaves para preparar o ambiente do IBM WebSphere MQ, mesmo se você não pretender armazenar certificados de autoridade de certificação (CA) nele, mas irá armazenar

todos os certificados no seu hardware de criptografia. Um banco de dados de chaves é necessário para que o gerenciador de filas faça referência sem seu campo SSLKEYR ou para o aplicativo cliente fazer referência na variável de ambiente MQSSLKEYR. Este banco de dados de chaves também é necessário se você estiver criando uma solicitação de certificado.

Você cria o banco de dados de chaves usando a linha de comandos ou usando a interface com o usuário **strmqikm** (iKeyman).

Procedimento

Crie um banco de dados de chave usando a linha de comandos

1. Execute um dos comandos a seguir:
   - Nos sistemas UNIX, Linux, and Windows:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Usando runmqakm:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
         -stash -fips -strong
     ```

   em que:

   -db *filename*
   Especifica o nome completo do arquivo de um banco de dados de chaves do CMS e deve ter uma extensão de arquivo de `.kdb`.

   -pw *password*
   Especifica a senha para o banco de dados de chaves do CMS.

   -type *cms*
   Especifica o tipo de banco de dados. (Para IBM WebSphere MQ, ele deve ser `cms`.)

   -stash
   Salva a senha do banco de dados de chaves em um arquivo.

   -fips
   Desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando em modo FIPS, o componente ICC usa algoritmos que são validados para FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

   -strong
   Verifica se a senha inserida atende aos requisitos mínimos de força da senha. Os requisitos mínimos para uma senha são como a seguir:
   - A senha deve ter no mínimo 14 caracteres.
   - A senha deve conter no mínimo um caractere minúsculo, um caractere maiúsculo e um dígito ou caractere especial. Os caracteres especiais incluem o asterisco (*), o sinal de dólar ($), o sinal de número (#) e o sinal de percentual (%). Um espaço é classificado como um caractere especial.
   - Cada caractere pode ocorrer no máximo de três vezes em uma senha.
   - O máximo de dois caracteres consecutivos na senha podem ser idênticos.
   - Todos os caracteres estão no conjunto de caracteres padrão ASCII para impressão no intervalo 0x20 - 0x7E.

Como alternativa, crie um banco de dados de chaves usando a interface com o usuário **strmqikm** (iKeyman).

2. Nos sistemas UNIX and Linux, efetue login como usuário raiz. Nos sistemas Windows, efetue login como Administrador ou como um membro do grupo MQM.
3. Inicie a interface com o usuário do iKeyman executando o comando **strmqikm**.
4. Clique em **Arquivo do Banco de Chave** > **Abrir**.

5. Clique em **Tipo de banco de dados de chave** e selecione **PKCS11Direct**.
6. No campo **Nome do arquivo**, digite o nome do módulo para gerenciar o hardware de criptografia; por exemplo, `PKCS11_API.so`.

   Se você estiver usando certificados ou chaves armazenadas no hardware de criptografia PKCS #11, observe que iKeycmd e iKeyman são programas de 64 bits. Módulos externos necessários para o suporte do PKCS #11 serão carregados em um processo de 64 bits, portanto, você deverá ter uma biblioteca do PKCS #11 de 64 bits instalada para a administração do hardware de criptografia. As plataformas Windows e Linux x86 de 32 bits são as únicas exceções, pois os programas iKeyman e iKeycmd são de 32 bits nessas plataformas.
7. No campo **Localização**, insira o caminho:

   - Nos sistemas UNIX and Linux, este pode ser `/usr/lib/pksc11`, por exemplo.
   - Nos sistemas Windows, é possível digitar o nome da biblioteca; por exemplo, `cryptoki`.

   Clique em **OK**. A janela Abrir Token Criptográfico é aberta.
8. No campo **Senha de Token de Criptografia**, digite a senha que você definiu ao configurar o hardware de criptografia.
9. Se o seu hardware de criptografia possuir o recurso para manter os certificados do signatário necessários para receber ou importar um certificado pessoal, limpe ambas as caixas de opções do banco de dados de chave secundários e continue a partir da etapa “13” na página 143.

   Se você requerer um banco de dados de chave CMS secundário para manter os certificados de assinante, selecione **Abrir arquivo existente do banco de dados de chave secundário** ou **Criar novo arquivo secundário do banco de dados de chave**.
10. No campo **Nome do Arquivo**, digite um nome de arquivo. Esse campo já contém o texto `key.kdb`. Se o seu nome de raiz for `key`, deixe esse campo inalterado Se você especificou um nome de raiz diferente, substitua `key` pelo seu nome de raiz. Não se deve mudar o sufixo `.kdb`.
11. No campo **Localização** digite o caminho, por exemplo:

    - Para um gerenciador de filas: `/var/mqm/qmgrs/QM1/ssl`
    - Para um IBM WebSphere MQ cliente MQI: `/var/mqm/ssl`

    Clique em **OK**. A janela Prompt de Senha é aberta.
12. Insira uma senha.

    Se você selecionou **Abrir arquivo existente do banco de dados de chave secundário** na etapa “9” na página 143, digite uma senha no campo **Senha**.

    Se você selecionou **Criar novo arquivo do banco de dados de chave secundário** na etapa “9” na página 143; conclua as seguintes subetapas:

    a) Digite a senha no campo **Senha** e digite-a novamente no campo **Confirmar Senha**.

    b) Selecione **Criar um arquivo stash contendo a senha**. Observe que se você não proteger a senha, as tentativas para iniciar os canais SSL falharão porque não é possível obter a senha necessária para acessar o arquivo de banco de dados de chave.

    c) Clique em **OK**. Uma janela é aberta, confirmando que a senha está no arquivo `key.sth` (a menos que você tenha especificado um nome de raiz diferente).
13. Clique em **OK**. O quadro do conteúdo do banco de dados de Chave é exibido.

Solicitando um Certificado Pessoal para o Hardware PKCS #11

Use este procedimento para um gerenciador de filas ou um cliente MQI do IBM WebSphere MQ para solicitar um certificado pessoal para seu hardware de criptografia.

Usando a interface com o usuário do iKeyman

Sobre esta tarefa

Nota: WebSphere MQ não suporta algoritmos SHA-3 ou SHA-5 . É possível usar os nomes de algoritmos de assinatura digital SHA384WithRSA e SHA512WithRSA porque ambos os algoritmos são membros da família do SHA-2.

Os nomes dos algoritmos de assinatura digital SHA3WithRSA e SHA5WithRSA são descontinuados porque eles são uma forma abreviada do SHA384WithRSA e do SHA512WithRSA, respectivamente.

Procedimento

Para solicitar um certificado pessoal a partir da interface com o usuário do iKeyman, conclua as etapas a seguir:

1. Conclua as etapas para trabalhar com o hardware de criptografia. Consulte o "Gerenciando certificados em hardware PKCS #11" na página 141.
2. No menu **Criar** clique em **Novo Pedido de Certificado**.

   A janela Criar Nova Chave e Pedido de Certificado é aberta.
3. No campo **Rótulo chave** , insira os rótulos a seguir:
   - Para um gerenciador de filas, insira `ibmwebspheremq` seguido pelo nome de seu gerenciador de filas alterado para minúsculas. Por exemplo, para um gerenciador de filas chamado `QM1`, insira `ibmwebspheremqqm1`
   - Para um IBM WebSphere MQ MQI client, insira `ibmwebspheremq` seguido por seu ID do usuário de logon, tudo em letras minúsculas; por exemplo, `ibmwebspheremqmyuserid` .
4. Insira valores para **Nome Comum** e **Organização**e selecione um **País** . Para os campos opcionais remanescentes, você pode tanto aceitar os valores padrão como digitar ou selecionar novos valores.

   Observe que é possível fornecer apenas um nome no campo **Unidade Organizacional**. Para obter informações adicionais sobre estes campos, consulte "Nomes Distintos" na página 11.
5. No campo **Inserir o nome de um arquivo no qual armazenar a solicitação de certificado**, aceite o padrão `certreq.arm` ou insira um novo valor com um caminho completo.
6. Clique em **OK**.

   A janela de confirmação é aberta.
7. Clique em **OK**.

   A lista **Pedidos de Certificado Pessoal** mostra o rótulo do novo pedido de certificado pessoal que você criou. O pedido de certificado é armazenado no arquivo que você escolheu na etapa "5" na página 144.
8. Solicite o novo certificado pessoal enviando o arquivo para uma autoridade de certificação (CA) ou copiando o arquivo no formulário de solicitação no website para a CA.

Usando a linha de comandos

Procedimento

Use os comandos a seguir para solicitar um certificado pessoal usando o comando **runmqckm** ou o comando **runmqakm**:

- Usando o **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -sig_alg algorithm
```

Em vez de `-dn` *distinguished_name* , é possível usar `-san_dsname` *DNS_names* , `-san_emailaddr` *email_addresses* ou `-san_ipaddr` *IP_addresses* .

- Usando runmqakm:

```
runmqakm -certreq -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -file filename -fips
        -sig_alg algorithm
```

em que:

-db *filename*
Especifica o nome completo do arquivo de um banco de dados de chaves do CMS.

-pw *password*
Especifica a senha para o banco de dados de chaves do CMS.

-label *label*
Especifica o rótulo chave anexado ao certificado.

-dn *distinguished_name*
Especifica o nome distinto X.500 colocado entre aspas duplas. Pelo menos um atributo é necessário. É possível fornecer diversos atributos OU e DC.

-size *key_size*
Especifica o tamanho da chave. Se você estiver usando **runmqckm** , o valor poderá ser 512 ou 1024. Se estiver usando **runmqakm**, o valor pode ser 512, 1024 ou 2048.

-file *filename*
Especifica o nome do arquivo para a solicitação de certificado.

-fips
Especifica que o comando é executado no modo FIPS. Este modo desativa o uso da biblioteca de criptografia BSafe. Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando em modo FIPS, o componente ICC usa algoritmos que são validados para FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

-sig_alg
Para **runmqckm**, especifica o algoritmo de assinatura assimétrica usado para a criação do par de chaves da entrada. The value can be 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, or SHAWithRSA . O valor padrão é SHA1WithRSA

-sig_alg
Para o **runmqakm**, especifica o algoritmo hash usado durante a criação de uma solicitação de certificado. Este algoritmo hash é usado para criar a assinatura associada a solicitação de certificado criada recentemente. O valor pode ser 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. O valor padrão é SHA1WithRSA.

-san_dnsname *DNS_names*
Especifica uma lista de nomes DNS delimitada por vírgulas ou espaços para a entrada sendo criada.

-san_emailaddr *email_addresses*
Especifica uma lista de endereços de e-mail delimitada por vírgulas ou espaços para a entrada sendo criada.

-san_ipaddr *IP_addresses*
Especifica uma lista de endereços IP delimitada por vírgulas ou espaços para a entrada sendo criada.

Importando um certificado pessoal para o hardware PKCS #11
Use este procedimento para um gerenciador de filas ou um cliente MQI do IBM WebSphere MQ para importar um certificado pessoal para seu hardware criptográfico

Usando o iKeyman

Procedimento

Para solicitar um certificado pessoal a partir da interface com o usuário do iKeyman, conclua as etapas a seguir:

1. Conclua as etapas para trabalhar com o hardware de criptografia. Consulte o "Gerenciando certificados em hardware PKCS #11" na página 141.
2. Clique em **Receive**. A janela Receber Certificado de um Arquivo é aberta.
3. Selecione o **Tipo de dados** do novo certificado pessoal; por exemplo `Base64-encoded ASCII data` para um arquivo com a extensão `.arm`.
4. Digite o nome do arquivo do certificado para o novo certificado pessoal ou clique em **Procurar** para selecionar o nome e a localização.
5. Clique em **OK**. Se você já tiver um certificado pessoal no banco de dados de chave de uma janela será aberta, perguntando se você deseja configurar a chave que está incluindo como a chave padrão no banco de dados.
6. Clique em **Sim** ou **Não**. A janela Inserir um Rótulo é aberta.
7. Digite um rótulo..

   Por exemplo, você pode usar o mesmo rótulo quando solicitou o certificado pessoal. Observe que o rótulo deve estar no formato IBM WebSphere MQ correto:

   - Para um gerenciador de filas, `ibmwebspheremq` seguido pelo nome de seu gerenciador de filas em letras minúsculas Por exemplo, para um gerenciador de filas chamado QM1, o rótulo seria: `ibmwebspheremqqm1`.
   - Para um IBM WebSphere MQ cliente MQI, `ibmwebspheremq` seguido por seu ID do usuário de logon em minúsculas. Por exemplo, para um ID do usuário `MyUserID`, o rótulo seria: `ibmwebspheremqmyuserid`.
8. Clique em **OK**. A lista **Certificados Pessoais** mostra o rótulo do novo certificado pessoal que você incluiu. Este rótulo é formado ao incluir o token criptográfico antes do rótulo que você forneceu.

Usando a linha de comandos

Procedimento

Para solicitar um certificado pessoal a partir de uma linha de comandos, conclua as etapas a seguir:

1. Abra uma janela de comandos configurada para seu ambiente.
2. Insira o comando apropriado para seu sistema operacional e configuração:

   - Em sistemas Windows, UNIX and Linux , use um destes comandos:

     ```
     runmqckm -cert -receive -file filename -crypto path
     -tokenlabel hardware_token -pw hardware_password -format cert_format
     ```

     ```
     runmqakm -cert -receive -file filename -crypto path
     -tokenlabel hardware_token -pw hardware_password -format cert_format -fips
     ```

   em que:

   -file *filename*
   Especifica o nome completo do arquivo que contém o certificado pessoal.

   -crypto *path*
   Especifica o caminho completo para a biblioteca PKCS #11 fornecida com o hardware.

-tokenlabel *hardware_token*
Especifica o rótulo fornecido para a parte de armazenamento do hardware de criptografia durante a instalação.

-pw *hardware_password*
Especifica a senha para acesso ao hardware.

-format *cert_format*
Especifica o formato do certificado. O valor pode ser `ascii` para ASCII codificado na Base64 ou `binary` para dados DER binários. O padrão é ASCII.

-fips
Especifica que o comando é executado no modo FIPS. Esse modo desativa o uso da biblioteca criptográfica BSafe Apenas o componente ICC será usado e este componente será inicializado com sucesso no modo FIPS. Quando em modo FIPS, o componente ICC usa algoritmos que são validados para FIPS 140-2. Se o componente ICC não for inicializado no modo FIPS, o comando **runmqakm** falhará.

Identificando e autenticando usuários

É possível identificar e autenticar usuários usando a estrutura MQCSP ou em diversos tipos de programa de saída de usuário.

Usando a estrutura MQCSP

É possível especificar a estrutura de parâmetros de segurança de conexão do MQCSP em uma chamada MQCONNX; essa estrutura contém um ID do usuário e senha. Se necessário, é possível mudar o MQCSP em uma saída de segurança.

Nota: O gerenciador de autoridade de objeto (OAM) não usa a senha. No entanto, o OAM faz algum trabalho limitado com o ID do usuário, que pode ser considerado uma forma comum de autenticação. Essas verificações param você adotar outro ID do usuário, se você usar esses parâmetros em seus aplicativos.

Implementando identificação e autenticação em saídas de segurança

O principal objetivo da saída de segurança é permitir que o MCA de cada extremidade de um canal autentique o seu parceiro. Em cada extremidade de um canal de mensagens, e na extremidade do servidor de um canal MQI, um MCA geralmente age em lugar do gerenciador de filas ao qual está conectado. Na extremidade do cliente de um canal MQI, um MCA geralmente age em nome do usuário do aplicativo cliente WebSphere MQ . Nessa situação, a autenticação mútua realmente ocorre entre dois gerenciadores de fila ou entre um gerenciador de fila e o usuário de um aplicativo cliente WebSphere MQ .

A saída de segurança fornecida (a saída de canal SSPI) ilustra como a autenticação mútua pode ser implementada trocando-se os tokens de autenticação que são gerados e, em seguida, verificados por um servidor de autenticação confiável, como o Kerberos. Para saber detalhes adicionais, consulte a seção “O Programa de Saída de Canal da SSPI” na página 105.

A autenticação mútua pode ser implementa também pela tecnologia PKI (Public Key Infrastructure). Cada saída de segurança gera alguns dados aleatórios, assina-os utilizando a tecla privativa do gerenciador ou usuário de fila que está representando e envia os dados assinados a seu parceiro em uma mensagem de segurança. A saída de segurança do parceiro executa a autenticação verificando a assinatura digital com a tecla privativa do gerenciador ou usuário da fila. Antes de trocar as assinaturas digitais, as saídas de segurança podem ter que concordar o algoritmo para gerar uma compilação de mensagem, se existir mais de um algoritmo disponível para uso.

Quando uma saída de segurança envia os dados assinados a seu parceiro, também precisa enviar algum meio de identificar o gerenciador ou usuário da fila que está representando. Pode ser um Nome Distinto ou mesmo um certificado digital. Se for enviado um certificado digital, a saída de segurança do parceiro poderá validar o certificado, operando pela cadeia de certificados do certificado CA raiz. Isto garante a propriedade da tecla pública utilizada para verificar a assinatura digital.

A saída de segurança do parceiro poderá validar um certificado digital somente se tiver acesso a um repositório de chaves que contenha os certificados restantes na cadeia de certificados. Se um certificado digital do gerenciador ou usuário de fila não for enviado, deverá haver um disponível no repositório de chaves ao qual a saída de segurança do parceiro tenha acesso. A saída de segurança do parceiro não poderá verificar a assinatura digital a menos que encontre a tecla pública do assinante.

O Secure Sockets Layer (SSL) e a Segurança da Camada de Transporte (TLS) usam técnicas de PKI, como as recém-descritas. Para obter mais informações sobre como SSL e TLS executam a autenticação, consulte “Conceitos do Secure Sockets Layer (SSL) e Transport Layer Security (TLS)” na página 14.

Se o suporte do servidor de autenticação ou PKI não estiver disponível, poderão ser utilizadas outras técnicas. Uma técnica comum, que pode ser implementada em saídas de segurança, utiliza um algoritmo de chave simétrico.

Uma das saídas de segurança, saída A, gera um número aleatório e o envia em uma mensagem de segurança para sua saída de segurança parceira, a saída B. A saída B criptografa o número usando sua cópia de uma chave que é conhecida apenas pelas duas saídas de segurança. A saída B envia o número criptografado para a saída A em uma mensagem de segurança com um segundo número aleatório que a saída B gerou. A saída A verifica se o primeiro número aleatório foi criptografado corretamente, criptografa o segundo número aleatório utilizando sua cópia da chave e envia o número criptografado à saída B em uma mensagem de segurança. A saída B verifica então se o segundo número foi criptografado corretamente. Durante essa troca, se nenhuma saída de segurança estiver satisfeita com a autenticidade da outra, poderá instruir o MCA a fechar o canal.

Uma vantagem desta técnica é que nenhuma chave ou senha é enviada para a conexão de comunicações durante a troca. Uma desvantagem é que não fornece uma solução para o problema de como distribuir a chave compartilhada de forma segura. Uma solução para esse problema está descrita em “Implementando confidencialidade em programas de saída do usuário” na página 229. Uma técnica semelhante é utilizada no SNA para a autenticação mútua de duas LUs quando elas se ligam para formar uma sessão. A técnica está descrita no “Autenticação em nível de sessão” na página 76.

Todas as técnicas anteriores de autenticação mútua podem ser adaptadas para fornecer autenticação unilateral.

Implementando a identificação e autenticação em saídas de mensagem

Quando um aplicativo põe uma mensagem em uma fila, o campo *UserIdentifier* no descritor da mensagem contém um ID do usuário associado ao aplicativo. Porém, não há dados presentes para serem utilizados para autenticar o ID do usuário. Esses dados podem ser incluídos por uma saída de mensagem na extremidade de envio de um canal e verificados por uma saída de mensagem na extremidade receptora do canal. Os dados de autenticação podem ser uma senha criptografada ou uma assinatura digital, por exemplo.

Este serviço pode ser mais efetivo se implementado no nível do aplicativo. O requisito básico é que o usuário do aplicativo que recebe a mensagem seja capaz de identificar e autenticar o usuário do aplicativo que enviou a mensagem. É, portanto, natural considerar a implementação desse serviço no nível do aplicativo. Para obter informações adicionais, consulte “Mapeamento de identidade na saída de API e saída cruzada da API” na página 151.

Implementando identificação e autenticação na saída de API e saída cruzada da API

No nível de uma mensagem individual, identificação e autenticação é um serviço que envolve dois usuários, o emissor e o receptor da mensagem. O requisito básico é que o usuário do aplicativo que recebe a mensagem seja capaz de identificar e autenticar o usuário do aplicativo que enviou a mensagem. Observe que este requisito serve para autenticação de uma via, não de duas vias.

Dependendo de como seja implementado, os usuários e seus aplicativos podem precisar fazer interface, ou mesmo interagir, com o serviço. Além disso, quando e como o serviço será utilizado pode depender de onde os usuários e seus aplicativos estão localizados, e da natureza dos próprios aplicativos. É, portanto, natural considerar a implementação do serviço ao nível do aplicativo, ao invés de ao nível do link.

Se você considerar a implementação deste serviço ao nível do link, você pode precisar resolver questões tais como as seguintes:

- Em um canal de mensagens, como aplicar o serviço apenas às mensagens que precisam?
- Como habilitar usuários e seus aplicativos a fazer interface, ou interagir, com o serviço, se isso é um requisito?
- Em uma situação de multisalto, em que uma mensagem é enviada em mais de um canal de mensagens a caminho de seu destino, onde você chamará os componentes do serviço?

Aqui estão alguns exemplos de como o serviço de identificação e autenticação pode ser implementado no nível do aplicativo. O termo *saída API* significa tanto uma saída API, quanto uma saída cruzada da API.

- Quando um aplicativo coloca uma mensagem em uma fila, uma saída API pode adquirir um token de autenticação de um servidor de autenticação confiável, como Kerberos. A saída API pode incluir este token nos dados do aplicativo na mensagem. Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode pedir ao servidor de autenticação que autentique o emissor verificando o token.
- Quando um aplicativo põe uma mensagem em uma fila, uma saída API pode anexar os seguintes itens aos dados do aplicativo na mensagem:
  - O certificado digital do emissor
  - A assinatura digital do emissor

  Se diferentes algoritmos para gerar uma compilação da mensagem estiverem disponíveis para utilização, a saída API pode incluir o nome do algoritmo que ela utilizou.

  Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode executar as seguintes verificações:

  - A saída API pode validar o certificado digital verificando toda a cadeia de certificados até o certificado de CA raiz. Para fazer isto a saída da API deve ter acesso ao repositório de chaves que contém os certificados restantes na cadeia de certificados. Esta verificação proporciona garantia de que o emissor, identificado pelo Nome Distinto, seja o proprietário genuíno da chave pública contida no certificado.
  - A saída API pode verificar a assinatura digital utilizando a chave pública contida no certificado. Essa verificação autentica o emissor.

  O Nome Distinto do emissor pode ser enviado, ao invés do certificado digital inteiro. Neste caso, o repositório de chaves deve conter o certificado do emissor, de modo que a segunda saída API possa encontrar a chave pública do emissor. Outra possibilidade é enviar todos os certificados na cadeia de certificados.
- Quando um aplicativo põe uma mensagem em uma fila, o campo *UserIdentifier* no descritor da mensagem contém um ID do usuário associado ao aplicativo. O ID do usuário pode ser utilizado para identificar o emissor. Para ativar a autenticação, uma saída API pode anexar alguns dados, tal com uma senha criptografada, aos dados do aplicativo na mensagem. Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode autenticar o ID do usuário utilizando os dados que seguiram com a mensagem.

  Esta técnica pode ser considerada suficiente para mensagens que se originem em um ambiente controlado e confiável, e em circunstâncias em que um servidor de autenticação confiável ou suporte PKI não estejam disponíveis.

Usuários Privilegiados

Um usuário privilegiado é aquele que possui autoridades administrativas completas para WebSphere MQ.

Além dos usuários listados na tabela a seguir, os membros de qualquer grupo com autoridade `+crt` para as filas indiretamente são administradores. Da mesma forma, qualquer usuário que tenha autoridade `+set` no gerenciador de filas e a autoridade `+put` na fila de comandos é um administrador.

Não é necessário conceder esses privilégios a usuários e aplicativos ordinários.

Tabela 13. Usuários privilegiados por plataforma.

Uma tabela de usuários privilegiados. No Windows, SYSTEM, todos os membros do grupo mqm e todos os membros do grupo Administradores são usuários privilegiados. Nos sistemas UNIX and Linux todos os membros do grupo mqm são usuários privilegiados. No IBM i, os perfis (usuários) qmqm e qmqmadm, todos membros do grupo qmqmadm e qualquer usuário definido com a configuração *ALLOBJ são usuários privilegiados.

Plataforma	**Usuários Privilegiados**
Sistemas Windows	• SISTEMA • Membros do grupo mqm • Membros do grupo Administradores
Sistemas UNIX and Linux	• Membros do grupo mqm

Identificando e autenticando usuários usando a estrutura MQCSP

É possível especificar a estrutura de parâmetros de segurança de conexão do MQCSP em uma chamada MQCONNX.

A estrutura de parâmetros de segurança de conexão do MQCSP contém um ID do usuário e senha, que o serviço de autorização pode usar para identificar e autenticar o usuário.

O componente de serviço de autorização fornecido com o IBM WebSphere MQ é chamado Gerenciador de Autoridade de Objeto (OAM). O OAM autoriza usuários com base no ID contido no MQCSP mas não valida a senha. É possível implementar a validação de senha no serviço de autorização usando saídas de cadeia com o OAM ou substituindo o OAM com um serviço de autorização alternativa.

É possível alterar o MQCSP em uma saída de segurança.

Implementando identificação e autenticação em saídas de segurança

É possível usar uma saída de segurança para implementar a autenticação unilateral ou mútua.

O principal objetivo da saída de segurança é permitir que o MCA de cada extremidade de um canal autentique o seu parceiro. Em cada extremidade de um canal de mensagens, e na extremidade do servidor de um canal MQI, um MCA geralmente age em lugar do gerenciador de filas ao qual está conectado. Na extremidade do cliente de um canal MQI, um MCA geralmente age em nome do usuário do aplicativo cliente MQI do WebSphere MQ . Nessa situação, a autenticação mútua realmente ocorre entre dois gerenciadores de fila ou entre um gerenciador de fila e o usuário de um aplicativo cliente WebSphere MQ .

A saída de segurança fornecida (a saída de canal SSPI) ilustra como a autenticação mútua pode ser implementada trocando-se os tokens de autenticação que são gerados e, em seguida, verificados por um servidor de autenticação confiável, como o Kerberos. Para obter mais detalhes, consulte "O Programa de Saída de Canal da SSPI" na página 105.

A autenticação mútua pode ser implementa também pela tecnologia PKI (Public Key Infrastructure). Cada saída de segurança gera alguns dados aleatórios, assina-os utilizando a tecla privativa do gerenciador ou usuário de fila que está representando e envia os dados assinados a seu parceiro em uma mensagem de segurança. A saída de segurança do parceiro executa a autenticação verificando a assinatura digital com a tecla privativa do gerenciador ou usuário da fila. Antes de trocar as assinaturas digitais, as saídas de segurança podem ter que concordar o algoritmo para gerar uma compilação de mensagem, se existir mais de um algoritmo disponível para uso.

Quando uma saída de segurança envia os dados assinados a seu parceiro, também precisa enviar algum meio de identificar o gerenciador ou usuário da fila que está representando. Pode ser um Nome Distinto ou mesmo um certificado digital. Se for enviado um certificado digital, a saída de segurança do parceiro poderá validar o certificado, operando pela cadeia de certificados do certificado CA raiz. Isto garante a propriedade da tecla pública utilizada para verificar a assinatura digital.

A saída de segurança do parceiro poderá validar um certificado digital somente se tiver acesso a um repositório de chaves que contenha os certificados restantes na cadeia de certificados. Se um certificado digital do gerenciador ou usuário de fila não for enviado, deverá haver um disponível no repositório de chaves ao qual a saída de segurança do parceiro tenha acesso. A saída de segurança do parceiro não poderá verificar a assinatura digital a menos que encontre a tecla pública do assinante.

O Secure Sockets Layer (SSL) e a Segurança da Camada de Transporte (TLS) usam técnicas de PKI, como as recém-descritas. Para obter mais informações sobre como o SSL realiza a autenticação, consulte "Conceitos do Secure Sockets Layer (SSL) e Transport Layer Security (TLS)" na página 14.

Se o suporte do servidor de autenticação ou PKI não estiver disponível, poderão ser utilizadas outras técnicas. Uma técnica comum, que pode ser implementada em saídas de segurança, utiliza um algoritmo de chave simétrico.

Uma das saídas de segurança, saída A, gera um número aleatório e o envia em uma mensagem de segurança para sua saída de segurança parceira, a saída B. A saída B criptografa o número usando sua cópia de uma chave que é conhecida apenas pelas duas saídas de segurança. A saída B envia o número criptografado para a saída A em uma mensagem de segurança com um segundo número aleatório que a saída B gerou. A saída A verifica se o primeiro número aleatório foi criptografado corretamente, criptografa o segundo número aleatório utilizando sua cópia da chave e envia o número criptografado à saída B em uma mensagem de segurança. A saída B verifica então se o segundo número foi criptografado corretamente. Durante essa troca, se nenhuma saída de segurança estiver satisfeita com a autenticidade da outra, poderá instruir o MCA a fechar o canal.

Uma vantagem desta técnica é que nenhuma chave ou senha é enviada para a conexão de comunicações durante a troca. Uma desvantagem é que não fornece uma solução para o problema de como distribuir a chave compartilhada de forma segura. Uma solução para esse problema está descrita em "Implementando confidencialidade em programas de saída do usuário" na página 229. Uma técnica semelhante é utilizada no SNA para a autenticação mútua de duas LUs quando elas se ligam para formar uma sessão. A técnica está descrita no "Autenticação em nível de sessão" na página 76.

Todas as técnicas anteriores de autenticação mútua podem ser adaptadas para fornecer autenticação unilateral.

Mapeamento de identidade em saídas de mensagem

É possível usar saídas de mensagens para processar informações para autenticar um ID do usuário, mas pode ser melhor implementar a autenticação no nível do aplicativo.

Quando um aplicativo põe uma mensagem em uma fila, o campo *UserIdentifier* no descritor da mensagem contém um ID do usuário associado ao aplicativo. Porém, não há dados presentes para serem utilizados para autenticar o ID do usuário. Esses dados podem ser incluídos por uma saída de mensagem na extremidade de envio de um canal e verificados por uma saída de mensagem na extremidade receptora do canal. Os dados de autenticação podem ser uma senha criptografada ou uma assinatura digital, por exemplo.

Este serviço pode ser mais efetivo se implementado no nível do aplicativo. O requisito básico é que o usuário do aplicativo que recebe a mensagem seja capaz de identificar e autenticar o usuário do aplicativo que enviou a mensagem. É, portanto, natural considerar a implementação desse serviço no nível do aplicativo. Para obter mais informações, consulte "Mapeamento de identidade na saída de API e saída cruzada da API" na página 151.

Mapeamento de identidade na saída de API e saída cruzada da API

Um aplicativo que recebe uma mensagem deve ser capaz de identificar e autenticar o usuário do aplicativo que enviou a mensagem. Esse serviço é geralmente melhor implementado no nível do aplicativo. Saídas de API podem implementar o serviço de várias maneiras.

No nível de uma mensagem individual, identificação e autenticação é um serviço que envolve dois usuários, o emissor e o receptor da mensagem. O requisito básico é que o usuário do aplicativo que recebe a mensagem seja capaz de identificar e autenticar o usuário do aplicativo que enviou a mensagem. Observe que este requisito serve para autenticação de uma via, não de duas vias.

Dependendo de como seja implementado, os usuários e seus aplicativos podem precisar fazer interface, ou mesmo interagir, com o serviço. Além disso, quando e como o serviço será utilizado pode depender de onde os usuários e seus aplicativos estão localizados, e da natureza dos próprios aplicativos. É, portanto, natural considerar a implementação do serviço ao nível do aplicativo, ao invés de ao nível do link.

Se você considerar a implementação deste serviço ao nível do link, você pode precisar resolver questões tais como as seguintes:

- Em um canal de mensagens, como aplicar o serviço apenas às mensagens que precisam?
- Como habilitar usuários e seus aplicativos a fazer interface, ou interagir, com o serviço, se isso é um requisito?
- Em uma situação de multisalto, em que uma mensagem é enviada em mais de um canal de mensagens a caminho de seu destino, onde você chamará os componentes do serviço?

Aqui estão alguns exemplos de como o serviço de identificação e autenticação pode ser implementado no nível do aplicativo. O termo *saída API* significa tanto uma saída API, quanto uma saída cruzada da API.

- Quando um aplicativo coloca uma mensagem em uma fila, uma saída API pode adquirir um token de autenticação de um servidor de autenticação confiável, como Kerberos. A saída API pode incluir este token nos dados do aplicativo na mensagem. Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode pedir ao servidor de autenticação que autentique o emissor verificando o token.
- Quando um aplicativo põe uma mensagem em uma fila, uma saída API pode anexar os seguintes itens aos dados do aplicativo na mensagem:
  - O certificado digital do emissor
  - A assinatura digital do emissor

  Se diferentes algoritmos para gerar uma compilação da mensagem estiverem disponíveis para utilização, a saída API pode incluir o nome do algoritmo que ela utilizou.

  Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode executar as seguintes verificações:

  - A saída API pode validar o certificado digital verificando toda a cadeia de certificados até o certificado de CA raiz. Para fazer isto a saída da API deve ter acesso ao repositório de chaves que contém os certificados restantes na cadeia de certificados. Esta verificação proporciona garantia de que o emissor, identificado pelo Nome Distinto, seja o proprietário genuíno da chave pública contida no certificado.
  - A saída API pode verificar a assinatura digital utilizando a chave pública contida no certificado. Essa verificação autentica o emissor.

  O Nome Distinto do emissor pode ser enviado, ao invés do certificado digital inteiro. Neste caso, o repositório de chaves deve conter o certificado do emissor, de modo que a segunda saída API possa encontrar a chave pública do emissor. Outra possibilidade é enviar todos os certificados na cadeia de certificados.
- Quando um aplicativo põe uma mensagem em uma fila, o campo *UserIdentifier* no descritor da mensagem contém um ID do usuário associado ao aplicativo. O ID do usuário pode ser utilizado para identificar o emissor. Para ativar a autenticação, uma saída API pode anexar alguns dados, tal com uma senha criptografada, aos dados do aplicativo na mensagem. Quando a mensagem for recuperada pelo aplicativo receptor, uma segunda saída API pode autenticar o ID do usuário utilizando os dados que seguiram com a mensagem.

  Esta técnica pode ser considerada suficiente para mensagens que se originem em um ambiente controlado e confiável, e em circunstâncias em que um servidor de autenticação confiável ou suporte PKI não estejam disponíveis.

Trabalhando com Certificados Revogados

Os certificados digitais podem ser revogados pelas Autoridades de certificação. É possível verificar o status de revogação de certificados que usam OCSP ou CRLs nos servidores LDAP, dependendo da plataforma.

Durante o protocolo de reconhecimento do SSL, os parceiros de comunicação se autenticam com certificados digitais. A autenticação pode incluir a verificação de que o certificado recebido ainda pode ser confiável. As Autoridades de certificação (CAs) revogam certificados por muitas razões, incluindo:

- O proprietário mudou para uma organização diferente.
- A chave privada não é mais secreta

As CAs publicam os certificados pessoais revogados em uma CRL (Lista de Certificados Revogados). Os certificados de CA que foram revogados são publicados em uma ARL (Lista de Autoridades Revogadas).

Nos sistemas UNIX, Linux e Windows , o suporte SSL do WebSphere MQ verifica certificados revogados usando OCSP (Online Certificate Status Protocol) ou CRLs e ARLs em servidores LDAP (Lightweight Directory Access Protocol). OCSP é o método preferido. As classes IBM WebSphere MQ classes for Java e IBM WebSphere MQ classes for JMS não podem usar as informações de OCSP em um arquivo de tabela de definição de canal do cliente. No entanto, você pode configurar o OCSP conforme descrito na seção Usando Online Certificate Protocol.

Em z / Os e IBM i WebSphere MQ , o suporte SSL verifica certificados revogados usando CRLs e ARLs somente em servidores LDAP

Para obter mais informações sobre Certificado

Autoridades, consulte “Certificados Digitais” na página 9..

Certificados Revogados e OCSP

O IBM WebSphere MQ determina qual respondente Online Certificate Status Protocol (OCSP) usar e manipula a resposta recebida. Pode ser necessário concluir etapas para tornar o respondente do OCSP acessível.

Nota: Estas informações se aplicam apenas ao WebSphere MQ em sistemas Windows, UNIX and Linux.

Para verificar o status da revogação de um certificado digital usando OCSP, o WebSphere MQ pode usar dois métodos para determinar o contato com o respondente do OCSP:

- Usando a extensão de certificado AuthorityInfoAccess (AIA) no certificado a ser verificado.
- Usando uma URL especificada em um objeto de informação de autenticação ou especificada por um aplicativo cliente.

Uma URL especificada em um objeto de informações de autenticação ou por um aplicativo cliente que tem prioridade sobre uma URL em uma extensão de certificado de AIA.

Se a URL do respondente do OCSP estiver atrás de um firewall, reconfigure o firewall para que o respondente do OCSP possa ser acessado ou configure um servidor proxy do OCSP. Especifique o nome do servidor proxy usando a variável SSLHTTPProxyName na sub-rotina SSL. Nos sistemas do cliente, também é possível especificar o nome do servidor proxy usando a variável de ambiente MQSSLPROXY. Para obter mais detalhes, consulte as informações relacionadas.

Se você não estiver preocupado se os certificados TLS ou SSL estão revogados, talvez porque esteja em um ambiente de teste, poderá configurar OCSPCheckExtensions para NO na sub-rotina SSL. Se você configurar essa variável, qualquer extensão de certificado AIA será ignorada. É possível que essa solução não seja aceita em um ambiente de produção, no qual você pode querer não permitir o acesso de usuários que possuírem certificados revogados.

A chamada para acessar o respondente do OCSP pode resultar em um dos três resultados a seguir:

válido
O certificado é válido.

Revogado
O certificado é revogado.

Desconhecido.
Esse resultado pode surgir por um dos três motivos:

- O IBM WebSphere MQ não pode acessar o respondente OCSP.
- O respondente do OCSP enviou uma resposta, mas o WebSphere MQ não pode verificar a assinatura digital da resposta.
- O respondente do OCSP enviou uma resposta indicando que ele não possui nenhum dado de revogação para o certificado.

Se o IBM WebSphere MQ receber um resultado do OCSP de `Desconhecido`, seu comportamento dependerá da configuração do atributo OCSPAuthentication. Para gerenciadores de filas, este atributo fica retido na sub-rotina SSL do arquivo `qm.ini` para sistemas UNIX and Linux ou no registro do Windows. Ele pode ser configurado usando o IBM WebSphere MQ Explorer. Para os clientes, isso é mantido na sub-rotina SSL do arquivo de configuração do cliente.

Se o resultado `Desconhecido` for recebido e uma OCSPAuthentication for configurada para REQUIRED (o valor padrão), o WebSphere MQ rejeitará a conexão e emitirá uma mensagem de erro do tipo AMQ9716. Se as mensagens de evento do SSL do gerenciador de filas estiverem ativadas, uma mensagem de evento SSL do tipo MQRC_CHANNEL_SSL_ERROR com ReasonQualifier configurado para MQRQ_SSL_HANDSHAKE_ERROR é gerada.

Se o resultado `Desconhecido` for recebido e uma OCSPAuthentication for configurada para OPTIONAL, o WebSphere MQ permitirá que o canal SSL seja iniciado e nenhum aviso ou mensagem do evento do SSL será gerada.

Se um resultado `Desconhecido` for recebido e OCSPAuthentication estiver configurado como WARN, o canal SSL será iniciado, mas o IBM WebSphere MQ emitirá uma mensagem de aviso do tipo AMQ9717 no log de erro. Se as mensagens de evento do SSL do gerenciador de filas estiverem ativadas, uma mensagem de evento SSL do tipo MQRC_CHANNEL_SSL_WARNING com ReasonQualifier configurado para MQRQ_SSL_UNKNOWN_REVOCATION é gerada.

Assinatura Digital das Respostas do OCSP

Um respondente do OCSP pode assinar suas respostas de uma de três maneiras. Seu respondente informará qual método é usado.

- A resposta do OCSP pode ser assinada digitalmente usando o mesmo certificado de CA que emitiu o certificado que estiver verificando. Nesse caso, não é necessário configurar nenhum certificado adicional; as etapas já concluídas para estabelecer a conectividade SSL são suficientes para verificar a resposta do OCSP.
- A resposta do OCSP pode ser assinada digitalmente usando outro certificado assinado pela mesma autoridade de certificação (CA) que emitiu o certificado que você está verificando. O certificado de assinatura é enviado junto com a resposta do OCSP nesse caso. O certificado que fluiu do respondente OCSP deve ter uma Extensão de Uso de Chave Estendida configurada como `id-kp-OCSPSigning` para que possa ser confiável para este propósito. Como a resposta OCSP é enviada com o certificado que a assinou (certificado que seja assinado por uma CA que já é confiável para a conectividade do SSL), nenhuma configuração de certificado adicional é necessária.
- A resposta do OCSP pode ser assinada digitalmente usando outro certificado que não esteja relacionado diretamente ao certificado que estiver verificando. Neste caso, a resposta do OCSP é assinada por um certificado emitido pelo próprio respondente do OCSP. Você deve incluir uma cópia do certificado do respondente OCSP no banco de dados de chaves do cliente ou gerenciador de filas que executa a verificação do OCSP; consulte “Incluindo um certificado de autoridade de certificação (ou a parte pública de um certificado autoassinado) em um repositório de chaves, em sistemas UNIX, Linux, and Windows” na página 134. Quando um certificado de CA é incluído, por padrão, ele é incluído como uma raiz confiável, que é a configuração necessária nesse contexto. Se esse certificado não for incluído, o WebSphere MQ não poderá verificar a assinatura digital na resposta OCSP e a verificação

OCSP resultará em um resultado Desconhecido, podendo fazer com que o IBM WebSphere MQ feche o canal, dependendo do valor de OCSPAuthentication.

Online Certificate Status Protocol (OCSP) em aplicativos cliente Java e JMS

Devido a uma limitação da API Java, o WebSphere MQ pode utilizar a verificação de revogação do OCSP (Online Certificate Status Protocol) para soquetes seguros SSL e TLS somente quando OCSP está ativado para todo o processo da Java virtual machine (JVM). Existem duas maneiras de ativar o OCSP para todos os soquetes seguros na JVM:

- Edite o arquivo JRE java.security para incluir as definições de configuração do OCSP mostradas na Tabela 1 e reinicie o aplicativo.
- Use a API java.security.Security.setProperty(), sujeita a qualquer política do Gerenciador de Segurança Java em efeito.

No mínimo, é necessário especificar um dos valores ocsp.enable e ocsp.responderURL.

Nome da Propriedade	Descrição
ocsp.enable	Este valor da propriedade é `true` ou `false`. Se `true`, a verificação de OCSP é ativada ao realizar a verificação de revogação de certificados; se `false` ou não configurado, a verificação de OCSP está desativada.
ocsp.responderURL	Este valor de propriedade é uma URL que identifica o local do respondente do OCSP. Aqui está um exemplo; `ocsp.responderURL=http://ocsp.example.net:80`. Por padrão, o local do OCSP respondente é determinado implicitamente a partir do certificado que estiver sendo validado. A propriedade será usada quando a extensão Acesso de Informações de Autoridade (definida na RFC 3280) estiver ausente do certificado ou quando requerer substituição.
ocsp.responderCertSubjectName	Este valor da propriedade é o nome do assunto do certificado do respondente do OCSP. Aqui está um exemplo; `ocsp.responderCertSubjectName="CN=OCSP Responder, O=XYZ Corp"`. Por padrão, o certificado do respondente do OCSP é aquele do emissor do certificado que está sendo validado. Esta propriedade identifica o certificado do respondente do OCSP quando o padrão não se aplica. Seu valor é um nome distinto de sequência (definido no RFC 2253) que identifica um certificado no conjunto de certificados que são fornecidos durante a validação do caminho do certificado. Nos casos em que o nome do assunto sozinho não é suficiente para identificar exclusivamente o certificado, as propriedades ocsp.responderCertIssuerName e ocsp.responderCertSerialNumber devem ser usadas. Quando esta propriedade for definida, então as propriedades ocsp.responderCertIssuerName e ocsp.responderCertSerialNumber serão ignoradas.
ocsp.responderCertIssuerName	Este valor da propriedade é o nome do emissor do certificado do respondente do OCSP. Aqui está um exemplo; `ocsp.responderCertIssuerName="CN=Enterprise CA, O=XYZ Corp"`. Por padrão, o certificado do respondente do OCSP é aquele do emissor do certificado que está sendo validado. Esta propriedade identifica o certificado do respondente do OCSP quando o padrão não se aplica. Seu valor é um nome distinto de sequência (definido no RFC 2253) que identifica um certificado no conjunto de certificados que são fornecidos durante a validação do caminho do certificado. Quando esta propriedade for definida,

Nome da Propriedade	Descrição
	a propriedade ocsp.responderCertSerialNumber também deve ser configurada. Essa propriedade é ignorada quando a propriedade ocsp.responderCertSubjectName está configurada.
ocsp.responderCertSerialNumber	Este valor da propriedade é o número de série do certificado do respondente do OCSP. Aqui está um exemplo; `ocsp.responderCertSerialNumber=2A:FF:00`. Por padrão, o certificado do respondente do OCSP é aquele do emissor do certificado que está sendo validado. Esta propriedade identifica o certificado do respondente do OCSP quando o padrão não se aplica. Este valor é uma sequência de dígitos hexadecimais (separadores de dois pontos ou espaço podem estar presentes) que identifica um certificado no conjunto dos certificados fornecidos durante a validação do caminho do certificado. Quando esta propriedade for definida, a propriedade ocsp.responderCertIssuerName também deve ser configurada. Essa propriedade é ignorada quando a propriedade ocsp.responderCertSubjectName está configurada.

Antes de ativar do OCSP dessa forma, há várias considerações:

- Definir a configuração do OCSP afeta todos os soquetes seguros no processo da JVM. Em alguns casos, essa configuração pode ter efeitos colaterais indesejáveis quando a JVM é compartilhada com o código de outro aplicativo que usa soquetes seguros SSL ou TLS. Certifique-se de que a configuração do OCSP escolhido é adequada para todos os aplicativos que estão em execução na mesma JVM.
- Aplicar a manutenção para o seu JRE poderá sobrescrever o arquivo java.security. Tome cuidado ao aplicar as correções provisórias e manutenção do produto Java para evitar sobrescrever o arquivo java.security. Pode ser necessário reaplicar suas alterações java.security depois de aplicar a manutenção. Por essa razão, considere definir a configuração do OCSP usando a API java.security.Security.setProperty().
- Ativar a verificação de OCSP terá efeito somente se a verificação de revogação também estiver ativada. A verificação de revogação é ativada pelo método `PKIXParameters.setRevocationEnabled()`.
- Se estiver usando o Interceptor Java AMS descrito em Ativando a verificação de OCSP em interceptores nativos, tome cuidado para evitar usar uma configuração do OCSP java.security que entre em conflito com a configuração do OCSP AMS no arquivo de configuração de chaves.

Trabalhando com as Listas de Revogação de Certificados e Listas de Revogação de Autoridade

O suporte do WebSphere MQpara CRLs e ARLs varia por plataforma.

O suporte CRL e ARL em cada plataforma é o seguinte:

- No z/OS, o SSL do sistema suporta CRL e ARL armazenados em servidores LDAP pelo produto Tivoli Public Key Infrastructure
- Em outras plataformas, o suporte CRL e ARL é compatível com as recomendações de perfil PKIX X.509 V2 CRL.

O WebSphere MQ mantém um cache de CRLs e ARLs que foram acessados nas 12 horas anteriores

Quando um gerenciador de filas ou o cliente MQI do WebSphere MQ recebe um certificado, ele verifica a CRL para confirmar se o certificado ainda é válido WebSphere MQ primeiro efetua check-in do cache, se houver um cache. Se a CRL não estiver no cache, o WebSphere MQ interrogará os locais do servidor de CRL LDAP na ordem em que ocorrem na lista de nomes de objetos de informações sobre autenticação especificados pelo atributo *SSLCRLNamelist* , até que o WebSphere MQ localize uma CRL disponível. Se a lista de nomes não estiver especificada ou está especificada com um valor em branco, as CRLs não são verificadas.

Para obter mais informações sobre LDAP, consulte Usando serviços de protocolo de acesso de diretório leve com WebSphere MQ para Windows.

Configurando Servidores LDAP

Configure a Estrutura em Árvore de Informações do Diretório LDAP para que reflita a hierarquia de Nomes Distintos de CAs. Faça isso usando arquivos de Formato de Troca de Dados LDAP.

Configure a estrutura do LDAP DIT (Directory Information Tree) para utilizar a hierarquia correspondente aos Nomes Distintos das CAs que emitem certificados e CRLs. Você pode definir a estrutura DIT com um arquivo que utiliza o LDAP LDIF (Data Interchange Format). Você também pode utilizar arquivos do LDIF para atualizar um diretório.

Arquivos LDIF são arquivos de texto ASCII que contém as informações exigidas para definir objetos dentro do diretório LDAP. Os arquivos LDIF contêm uma ou mais entradas, cada uma das quais compreende um Nome Distinto, pelo menos uma definição de classe de objeto e, como opção, várias definições de atributos.

O atributo `certificateRevocationList;binary` contém uma lista em formato binário de certificados de usuários revogados. O atributo `authorityRevocationList;binary` contém uma lista binária de certificados CA que foram revogados. Para uso com SSL do WebSphere MQ , os dados binários para esses atributos devem estar em conformidade com o formato DER (Regras de Codificação Definidas). Para obter mais informações sobre os arquivos LDIF, consulte a documentação fornecida com o seu servidor LDAP.

Figura 12 na página 157 mostra um arquivo LDIF de amostra que você pode criar como entrada para seu servidor LDAP para carregar as CRLs e ARLs emitidas por CA1, que é uma Autoridade de Certificação imaginária com o Nome Distinto "CN=CA1, OU=Test, O=IBM, C=GB", configurado pela organização de Teste em 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)
```

Figura 12. Arquivo LDIF de amostra para uma Autoridade de certificação. Isso poderá variar de implementação para implementação.

Figura 13 na página 158 mostra a estrutura DIT que seu servidor LDAP cria quando você carrega o arquivo LDIF de amostra mostrado em Figura 12 na página 157 juntamente com um arquivo semelhante para CA2, uma autoridade de certificação imaginária configurada pela organização PKI, também dentro do IBM.

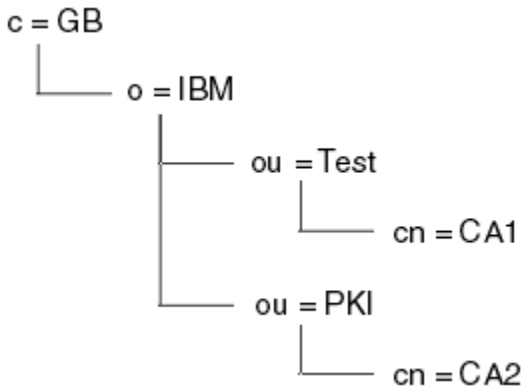


Figura 13. Exemplo de estrutura de Directory Information Tree do LDAP

WebSphere MQ verifica CRLs e ARLs.

Nota: Certifique-se de que a lista de controle de acesso para seu servidor LDAP permita que usuários autorizados leiam, pesquisem e comparem as entradas que contenham as CRLs e ARLs. O WebSphere MQ acessa o servidor LDAP usando as propriedades LDAPUSER e LDAPPWD do objeto AUTHINFO

Configurando e Atualizando Servidores LDAP
Use este procedimento para configurar ou atualizar o servidor LDAP.

1. Obtenha as CRLs e ARLs em formato DER a partir da sua Autoridade ou Autoridades de Certificação.
2. Utilizando o editor de texto ou a ferramenta fornecida em seu servidor LDAP, crie um ou mais arquivos LDIF que contenham o Nome Distinto da CA e as definições da classe de objetos exigidas. Copie os dados do formato DER para o arquivo LDIF como os valores do atributo `certificateRevocationList;binary` atributo para CRLs, o atributo `authorityRevocationList;binary` para ARLs ou ambos.
3. Inicie seu servidor LDAP.
4. Inclua as entradas do arquivo LDIF ou arquivos que você criou na etapa “2” na página 158.

Depois de configurar o servidor LDAP CRL, verifique se ele está configurado corretamente. Primeiro, tente utilizar um certificado que não esteja revogado no canal, e verifique se o canal foi iniciado corretamente. Em seguida utilize um certificado revogado e verifique se o canal falhou ao iniciar.

Obtenha regularmente as CRLs atualizadas a partir das Autoridades de certificação. Tente fazer esta operação em seus servidores LDAP a cada 12 horas.

Acessando as CRLs e ARLs com um Gerenciador de Filas

Um gerenciador de filas é associado a um ou mais objetos de informação de autenticação, que contêm o endereço de um servidor de LDAP CRL.

Observe que nesta seção, as informações sobre as CRLs (Certificate Revocation Lists) também se aplicam às ARLs (Authority Revocation Lists).

Informe ao gerenciador de filas como acessar as CRLs ao fornecê-los os objetos de informações de autenticação, cada qual mantendo o endereço de um servidor de CRL LDAP. Os objetos de informações de autenticação são mantidos em uma lista de nomes, que é especificada no atributo de gerenciador de filas *SSLCRLNamelist*.

No seguinte exemplo, o MQSC é utilizado para especificar os parâmetros:

1. Defina os objetos de informações de autenticação usando o comando DEFINE AUTHINFO MQSC com o parâmetro AUTHTYPE definido para CRLLDAP.

   O valor CRLLDAP para o parâmetro AUTHTYPE indica que as CRLs são acessadas em servidores LDAP. Cada objeto de informações de autenticação com o tipo CRLLDAP que você criar contém o endereço de um servidor LDAP. Quando você tem mais de um objeto de informações de autenticação,

os servidores LDAP os quais eles apontam *devem* conter informações idênticas. Isso fornece continuidade de serviço caso ocorra uma ou mais falhas nos servidores LDAP.

Em todas as plataformas, o ID do usuário e a senha são enviados não criptografados para o servidor LDAP.

2. Utilizando o comando DEFINE NAMELIST MQSC, defina uma lista de nomes para os nomes de seus objetos de informações de autenticação.
3. Utilizando o comando ALTER QMGR MQSC, forneça a lista de nomes ao gerenciador de filas. Por exemplo:

```
ALTER QMGR SSLCRLNL(sslcrlnlname)
```

em que `sslcrlnlname` é a sua lista de nomes de objetos de informações sobre autenticação

Este comando define um atributo de gerenciador de filas chamado *SSLCRLNamelist*. O valor inicial do gerenciador de filas para este atributo está em branco.

É possível incluir até 10 conexões para servidores LDAP alternativos para a lista de nomes, para garantir a continuidade do serviço se um ou mais servidores LDAP falharem. Observe que os servidores LDAP *devem* conter informações idênticas.

Acessando CRLs e ARLs usando o IBM WebSphere MQ Explorer
É possível usar o IBM WebSphere MQ Explorer para informar a um gerenciador de filas como acessar CRLs.

Observe que nesta seção, as informações sobre as CRLs (Certificate Revocation Lists) também se aplicam às ARLs (Authority Revocation Lists).

Utilize o seguinte procedimento para definir uma conexão LDAP para uma CRL:

1. Certifique-se de que iniciou o gerenciador de filas.
2. Clique com o botão direito na pasta **Informações sobre autenticação** e clique em **Novo -> Informações sobre autenticação**. Na folha de propriedade que abre:
   a. Na primeira página **Criar Informações sobre Autenticação** , insira um nome para o objeto CRL (LDAP).
   b. Na página **Geral** de **Alterar Propriedades**, selecione o tipo de conexão. Opcionalmente é possível digitar uma descrição.
   c. Selecione a página **CRL(LDAP)** de **Alterar Propriedades**.
   d. Digite o nome do servidor LDAP como o nome da rede ou o endereço IP.
   e. Caso o servidor exija os detalhes de login, forneça o ID do usuário e, se for preciso, uma senha.
   f. Clique em **OK**.
3. Clique com o botão direito na pasta **Listas de Nomes** e clique em **Novo-> Listas de Nomes** . Na folha de propriedade que abre:
   a. Digite um nome para a lista de nomes.
   b. Inclua o nome do objeto CRL(LDAP) (da etapa “2.a” na página 159) na lista.
   c. Clique em **OK**.
4. Clique com o botão direito do mouse no gerenciador de filas, selecione **Propriedades** e selecione a página **SSL**:
   a. Selecione a caixa de entrada **Verificar certificados recebidos por este gerenciador de filas comparandos às Listas de Certificados Revogados**.
   b. Digite o nome da lista de nomes (da etapa “3.a” na página 159) no campo **Nomes da CRL**.

Acessando CRLs e ARLs com um cliente MQI do IBM WebSphere MQ

Você tem três opções para especificar os servidores LDAP que contêm CRLs para verificação por um cliente MQI IBM WebSphere MQ .

Observe que nesta seção, as informações sobre as CRLs (Certificate Revocation Lists) também se aplicam às ARLs (Authority Revocation Lists).

As três maneiras de especificar os servidores LDAP são as seguintes:

- Utilizando uma tabela de definição de canais
- Usando a estrutura de opções de configuração do SSL, MQSCO, ou uma chamada de MQCONNX
- Usando o Active Directory (em sistemas Windows com suporte do Active Directory)

Para obter mais detalhes, consulte as informações relacionadas.

É possível incluir até 10 conexões para servidores LDAP alternativos, para garantir a continuidade do serviço se um ou mais servidores LDAP falharem. Observe que os servidores LDAP *devem* conter informações idênticas.

Não é possível acessar CRLs LDAP de um canal do cliente MQI do WebSphere MQ em execução na Linux (plataforma zSeries).

Local de um respondente OCSP e dos servidores LDAP que contêm CRLs
Em um sistema do cliente MQI do IBM WebSphere MQ , é possível especificar o local de um respondente do OCSP e de servidores Lightweight Directory Access Protocol (LDAP) que mantêm listas de revogação de certificado (CRLs)

É possível especificar estes locais de três maneiras, listadas aqui em ordem de precedência decrescente.

Quando um aplicativo cliente MQI do WebSphere MQ emite uma chamada MQCONNX

É possível especificar um respondente OCSP ou um servidor LDAP que contém CRLs em uma chamada **MQCONNX**.

Em uma chamada **MQCONNX**, a estrutura de opções de conexão, MQCNO, pode referenciar uma estrutura de opções de configuração de SSL, MQSCO. Por sua vez, a estrutura MQSCO pode referenciar uma ou mais estruturas de registro de informações sobre autenticação, MQAIR. Cada estrutura MQAIR contém todas as informações que um cliente MQI do WebSphere MQ requer para acessar um respondente OCSP ou um servidor LDAP contendo CRLs. Por exemplo, um dos campos em uma estrutura MQAIR é a URL na qual um replicador pode ser contatado. Para obter mais informações sobre a estrutura MQAIR, consulte MQAIR - Registro de informações sobre autenticação.

Usando uma tabela de definição de canal de cliente (ccdt) para acessar um respondente OCSP ou servidores LDAP

Para que um cliente MQI do WebSphere MQ possa acessar um respondente OCSP ou servidores LDAP que contêm CRLs, inclua os atributos de um ou mais objetos de informações sobre autenticação em uma tabela de definição de canal do cliente.

Em um gerenciador de filas do servidor é possível definir um ou mais objetos de informações sobre autenticação. Os atributos de um objeto de autenticação contêm todas as informações que são necessárias para acessar um respondente OCSP (em plataformas em que OCSP é suportado) ou um servidor LDAP que contém CRLs. Um dos atributos especifica a URL do respondente OCSP, outro especifica o endereço do host ou endereço IP de um sistema no qual um servidor LDAP é executado.

Um objeto de informações sobre autenticação com AUTHTYPE (OCSP) não se aplica para uso nos gerenciadores de filas IBM i ou z/OS , mas pode ser especificado nessas plataformas para ser copiado para a tabela de definição de canal do cliente (CCDT) para uso do cliente.

Para ativar um cliente MQI do WebSphere MQ para acessar um respondente OCSP ou servidores LDAP que contêm CRLs, os atributos de um ou mais objetos de informações sobre autenticação podem ser incluídos em uma tabela de definições de canal do cliente. É possível incluir atributos em uma das seguintes maneiras:

Nas plataformas do servidor AIX, HP-UX, Linux, Solaris e Windows

É possível definir uma lista de nomes que contém os nomes de um ou mais objetos de informações de autenticação. Você pode, então, configurar o atributo do gerenciador de filas, **SSLCRLNameList**com o nome desta lista de nomes.

Se você estiver usando CRLs, mais de um servidor LDAP poderá ser configurado para fornecer maior disponibilidade. A intenção é que cada servidor LDAP contenha as mesmas CRLs. Se um servidor LDAP estiver indisponível quando for necessário, um cliente MQI do WebSphere MQ poderá tentar acessar outro.

Os atributos dos objetos de informação de autenticação identificados pela lista de nomes são referidos coletivamente aqui como o *local de revogação de certificado*. Quando você configurar o atributo do gerenciador de filas, **SSLCRLNameList**, com o nome da lista de nomes, o local de revogação de certificado será copiado na tabela de definição de canal do cliente associada ao gerenciador de filas. Se o CCDT puder ser acessado a partir de um sistema do cliente como um arquivo compartilhado ou se o CCDT for copiado para um sistema do cliente, o cliente MQI do WebSphere MQ nesse sistema poderá usar o local de revogação de certificado no CCDT para acessar um respondente OCSP ou servidores LDAP que contêm CRLs.

Se o local de revogação de certificado do gerenciador de filas for mudado posteriormente, a mudança será refletida na CCDT associada ao gerenciador de filas. Se o atributo do gerenciador de filas, **SSLCRLNameList**, estiver configurado como em branco, o local de revogação de certificado será removido da CCDT. Estas alterações não são refletidas em nenhuma cópia da tabela em um sistema do cliente.

Se você requerer que o local de revogação de certificado nas extremidades do cliente e do servidor de um canal MQI seja diferente, e o gerenciador de filas do servidor for aquele que é usado para criar o local de revogação de certificado, será possível fazer isto conforme a seguir:

1. No gerenciador de filas do servidor, crie o local de revogação de certificado para uso no sistema do cliente.
2. Copie a CCDT contendo o local de revogação de certificado no sistema do cliente.
3. No gerenciador de filas do servidor, altere o local de revogação de certificado para o que é necessário na extremidade do servidor do canal MQI.

Usando Active Directory no Windows

Nos sistemas Windows , é possível usar o comando de controle **setmqcrl** para publicar as informações de CRL atuais no Active Directory

O comando **setmqcrl** não publica informações de OCSP.

Para obter informações sobre este comando e sua sintaxe, consulte setmqcrl.

Acessando CRLs e ARLs com classes IBM WebSphere MQ para Java e classes IBM WebSphere MQ para JMS

Classes IBM WebSphere MQ para Java e classes IBM WebSphere MQ para CRLs de acesso JMS de forma diferente de outras plataformas.

Para obter informações sobre como trabalhar com CRLs e ARLs com classes IBM WebSphere MQ para Java, consulte Usando listas de revogação de certificado.

Para obter informações sobre como trabalhar com CRLs e ARLs com classes IBM WebSphere MQ para JMS, consulte Propriedade do objeto SSLCERTSTORES..

Manipulando Objetos de Informações de Autenticação

É possível manipular objetos de informações sobre autenticação usando os comandos MQSC ou PCF ou o IBM WebSphere MQ Explorer.

Os seguintes comandos MQSC agem sobre objetos informações de autenticação:

- DEFINE AUTHINFO
- ALTER AUTHINFO
- DELETE AUTHINFO
- DISPLAY AUTHINFO

Para obter uma descrição completa desses comandos, consulte Comandos de Script (MQSC) .

Os seguintes comandos Programmable Command Format (PCF) agem sobre objetos de informações de autenticação:

- Criar Informações sobre Autenticação
- Copiar Informações sobre Autenticação
- Alterar Informações sobre Autenticação
- Excluir Informações sobre Autenticação
- Consultar Informações sobre Autenticação
- Consultar Nomes de Informações sobre Autenticação

Para obter uma descrição completa desses comandos, consulte Definições dos Formatos de Comando Programáveis .

Em plataformas nas quais ele está disponível, também é possível usar o WebSphere MQ Explorer

Autorizando o acesso aos objetos

Esta seção contém informações sobre como usar o gerenciador de autoridade de objeto e programas de saída de canal para controlar o acesso aos objetos.

Em sistemas UNIX, Linux, and Windows .... o acesso a objetos é controlado usando o gerenciador de autoridade de objeto (OAM). Esta coleção de tópicos contém informações sobre como usar a interface de comando para o OAM. Ele também contém uma lista de verificação que é possível usar para determinar quais tarefas executar para aplicar segurança a seu sistema e as considerações para conceder aos usuários a autoridade para administrar o IBM WebSphere MQ e trabalhar com objetos do IBM WebSphere MQ. Se os mecanismos de segurança fornecidas não atenderem suas necessidades, é possível desenvolver seus próprios programas de saída de canal.

Controlando o acesso a objetos usando o OAM nos sistemas UNIX, Linux e Windows

O object authority manager (OAM) fornece uma interface de comando para conceder e revogar autoridade para objetos do WebSphere MQ .

Você deve estar adequadamente autorizado para usar esses comandos, conforme descrito em “Autoridade para administrar IBM WebSphere MQ em sistemas UNIX, Linux, and Windows” na página 200. Os IDs de usuário que estão autorizados a administrar o WebSphere MQ possuem autoridade de *superusuário* para o gerenciador de filas, o que significa que você não precisa conceder a eles permissão adicional para emitir quaisquer solicitações MQI ou comandos.

Fornecendo acesso a um objeto IBM WebSphere MQ em sistemas UNIX, Linux, and Windows

Use o comando de controle **setmqaut** ou o comando PCF **MQCMD_SET_AUTH_REC** para fornecer aos usuários e grupos de usuários o acesso a objetos IBM WebSphere MQ

Para obter uma definição completa do comando de controle **setmqaut** e sua sintaxe, consulte setmqaute para obter uma definição completa do comando PCF **MQCMD_SET_AUTH_REC** e sua sintaxe, consulte Configurar Registro de Autoridade.

O gerenciador de filas deve estar em execução para usar esse comando. Quando você mudou o acesso de um diretor, as mudanças são refletidas imediatamente pelo OAM.

Para fornecer aos usuários acesso a um objeto, é necessário especificar:

- O nome do gerenciador de filas que possui os objetos com os quais você está trabalhando; se você não especificar o nome de um gerenciador de filas, o gerenciador de filas padrão será assumido.
- O nome e o tipo do objeto (para identificar o objeto exclusivamente). Especifique o nome como um *perfil*; este é o nome explícito do objeto ou um nome genérico, incluindo caracteres curinga. Para obter uma descrição detalhada de perfis genéricos e o uso de caracteres curinga dentro deles, consulte "Usando de perfis genéricos OAM em sistemas UNIX, Linux, and Windows" na página 164.
- Um ou mais nomes de principais e de grupos aos quais a autoridade se aplica.

  Se um ID do usuário contiver espaços, coloque-o entre aspas ao usar esse comando. Em sistemas Windows , é possível qualificar um ID do usuário com um nome de domínio Se o ID do usuário real contiver um símbolo de arroba (@), substitua-o por @@ para mostrar que ele faz parte do ID do usuário, não do delimitador entre o ID do usuário e o nome do domínio.
- Uma lista de autorizações. Cada item da lista especifica um tipo de acesso que deve ser concedido a esse objeto (ou revogado dele). Cada autorização na lista é especificada como uma palavra-chave, prefixada com um sinal de mais (+) ou um sinal de menos (-). Use um sinal de mais para incluir a autorização especificada, e um sinal de menos para remover a autorização. Não deve haver espaços entre os sinais de + ou - e a palavra-chave.

  É possível especificar qualquer número de autorizações em um único comando. Por exemplo, a lista de autorizações para permitir que um usuário ou um grupo coloque mensagens em uma fila e navegue por elas, mas revogue o acesso para obter mensagens é:

```
    +browse -get +put
```

Exemplos de Uso do Comando setmqaut

Os seguintes exemplos mostram como usar o comando `setmqaut` para conceder e revogar permissões de uso de um objeto:

```
setmqaut -m saturn.queue.manager -t queue -n RED.LOCAL.QUEUE
         -g groupa +browse -get +put
```

Nesse exemplo:

- `saturn.queue.manager` é o nome do gerenciador de filas
- `queue` é o tipo de objeto
- `RED.LOCAL.QUEUE` é o nome do objeto
- `groupa` é o identificador do grupo com autorizações que devem ser mudadas
- `+browse -get +put` é a lista de autorização para a fila especificada
  - `+browse` inclui autorização para navegar pelas mensagens na fila (para emitir **MQGET** com a opção de navegação)
  - `-get` remove a autorização para obter mensagens (**MQGET**) da fila
  - `+put` inclui autorização para colocar mensagens (**MQPUT**) na fila

O comando a seguir revoga a autoridade put na fila MyQueue do fvuser do diretor e dos grupos groupa e groupb. Em sistemas UNIX and Linux, este comando também revoga a autoridade put para todos os principais que estão no mesmo grupo primário que fvuser.

```
setmqaut -m saturn.queue.manager -t queue -n MyQueue -p fvuser
         -g groupa -g groupb -put
```

Usando o comando com um serviço de autorização diferente

Se você estiver usando seu próprio serviço de autorização em vez do OAM, é possível especificar o nome desse serviço no comando **setmqaut** para direcionar o comando para esse serviço. Você deverá

especificar esse parâmetro se tiver vários componentes instaláveis em execução ao mesmo tempo; caso contrário, a atualização será feita no primeiro componente instalável do serviço de autorização. Por padrão, esse é o OAM fornecido.

Usando de perfis genéricos OAM em sistemas UNIX, Linux, and Windows

Os perfis genéricos do OAM permitem configurar a autoridade que um usuário tem para muitos objetos de uma vez, em vez de ter que emitir comandos **setmqaut** separados em relação a cada objeto individual quando ele é criado.

Usar perfis genéricos no comando **setmqaut** permite configurar uma autoridade genérica para todos os objetos que se ajustam a esse perfil.

Esta coleção de tópicos descreve o uso de perfis genéricos em mais detalhes.

Usando Caracteres Curinga em Perfis OAM

O que torna um perfil genérico é o uso de caracteres especiais (caracteres curinga) no nome do perfil. Por exemplo, o caractere curinga de ponto de interrogação (?) corresponde a qualquer caractere único em um nome. Portanto, se você especificar `ABC.?EF`, a autorização concedida a esse perfil será aplicada a quaisquer objetos com os nomes `ABC.DEF` , `ABC.CEF`, `ABC.BEF`e assim por diante.

Os caracteres curinga disponíveis são:

?

Use o ponto de interrogação (?) em vez de qualquer caractere. Por exemplo, `AB.?D` se aplica aos objetos `AB.CD` , `AB.ED`e `AB.FD`.

*****

Use o asterisco (*) como:

- Um *qualificador* em um nome de perfil para corresponder a qualquer qualificador em um nome de objeto. Um qualificador é a parte de um nome de objeto delimitado por um ponto. Por exemplo, em `ABC.DEF.GHI`, os qualificadores são `ABC`, `DEF`e `GHI`

  Por exemplo, `ABC.*.JKL` se aplica aos objetos `ABC.DEF.JKL`e `ABC.GHI.JKL`. (Observe que ele **não** se aplica a `ABC.JKL`; * usado nesse contexto sempre indica um qualificador.)
- Um caractere dentro de um qualificador em um nome de perfil para corresponder a zero ou mais caracteres dentro do qualificador em um nome de objeto.

  Por exemplo, `ABC.DE*.JKL` se aplica aos objetos `ABC.DE.JKL`, `ABC.DEF.JKL`e `ABC.DEGH.JKL` .

******

Use o asterisco duplo (**) **uma vez** em um nome de perfil como:

- O nome do perfil inteiro para corresponder a todos os nomes de objeto. Por exemplo, se você usar `-t prcs` para identificar processos e, em seguida, usar ** como o nome do perfil, irá alterar as autorizações de todos os processos.
- Como o qualificador inicial, do meio ou final em um nome de perfil para corresponder a zero ou mais qualificadores em um nome de objeto. Por exemplo, o `**.ABC` identifica todos os objetos com o qualificador final ABC.

Nota: Ao usar caracteres curinga em sistemas UNIX and Linux **deve-se** colocar o nome de perfil entre aspas simples.

Prioridades do Perfil

Um ponto importante a ser entendido ao usar perfis genéricos é a prioridade que os perfis recebem ao decidirem quais autoridades aplicar a um objeto que está sendo criado. Por exemplo, suponha que você emitiu estes comandos:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

O primeiro fornece autoridade put para todas as filas para o fred principal com nomes que correspondem ao perfil AB. *; O segundo fornece autoridade get para os mesmos tipos de fila que correspondem ao perfil AB.C*.

Suponha que agora você crie uma fila chamada AB.CD. De acordo com as regras para correspondência de curinga, setmqaut poderia ser aplicado a essa fila. Portanto, ele tem autoridade put ou get?

Para localizar a resposta, você aplica a regra que, sempre que vários perfis puderem se aplicar a um objeto, **apenas o mais específico será aplicado**. A maneira de aplicação dessa regra é comparando-se os nomes dos perfis da esquerda para a direita. Sempre que forem diferentes, um caractere não genérico será mais específico do que um genérico. Portanto, no exemplo acima, a fila AB.CD tem autoridade **get** (AB.C* é mais específico que AB. *).

Ao comparar os caracteres genéricos, a ordem de *especificidade* é:

1. ?
2. *
3. **

Fazendo Dump de Configurações do Perfil

Para obter uma definição completa do comando de controle **dmpmqaut** e sua sintaxe, consulte dmpmqaute, para obter uma definição completa do comando PCF **MQCMD_INQUIRE_AUTH_RECS** e sua sintaxe, consulte Consultar Registros de Autoridade .

Os seguintes exemplos mostram o uso do comando de controle **dmpmqaut** para fazer dump de registros de autoridade de perfis genéricos:

1. Este exemplo faz dump de todos os registros de autoridade com um perfil que corresponde à fila a.b.c do principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   O dump resultante é semelhante a isto:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Nota: Embora os usuários do UNIX and Linux podem usar a opção `-p` para o comando **dmpmqaut**, eles devem usar `-g groupname` ao definir autorizações.
2. Este exemplo faz dump de todos os registros de autoridade com um perfil que corresponde à fila a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   O dump resultante é semelhante a isto:

   ```
   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. Este exemplo faz dump de todos os registros de autoridade para o perfil a.b. *, de fila do tipo

```
dmpmqaut -m qmgr1 -n a.b.* -t q
```

O dump resultante é semelhante a isto:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

4. Este exemplo faz dump de todos os registros de autoridade para o gerenciador de filas qmX.

```
dmpmqaut -m qmX
```

O dump resultante é semelhante a isto:

```
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. Este exemplo faz dump de todos os nomes de perfis e tipos de objetos para o gerenciador de filas qmX.

```
dmpmqaut -m qmX -l
```

O dump resultante é semelhante a isto:

```
profile: q1, type: queue
profile: q*, type: queue
profile: name.*, type: namelist
profile: pr1, type: process
```

Nota: Para WebSphere MQ para Windows apenas, todos os proprietários exibidos incluem informações de domínio, por exemplo:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

Usando Caracteres Curinga em Perfis OAM

Use caracteres curinga em um nome de perfil do Object Authority Manager (OAM) para que esse perfil seja aplicável a mais de um objeto.

O que torna um perfil genérico é o uso de caracteres especiais (caracteres curinga) no nome do perfil. Por exemplo, o caractere curinga de ponto de interrogação (?) corresponde a qualquer caractere único em um nome. Portanto, se você especificar `ABC.?EF`, a autorização fornecida a esse perfil se aplicará a qualquer objeto com os nomes `ABC.DEF`, `ABC.CEF`, `ABC.BEF`, etc.

Os caracteres curinga disponíveis são:

?

Use o ponto de interrogação (?) em vez de qualquer caractere. Por exemplo, `AB.?D` aplica-se aos objetos `AB.CD`, `AB.ED` e `AB.FD`.

*****

Use o asterisco (*) como:

- Um *qualificador* em um nome de perfil para corresponder a qualquer qualificador em um nome de objeto. Um qualificador é a parte de um nome de objeto delimitado por um ponto. Por exemplo, em `ABC.DEF.GHI`, os qualificadores são `ABC`, `DEF` e `GHI`.

  Por exemplo, `ABC.*.JKL` aplica-se aos objetos `ABC.DEF.JKL` e `ABC.GHI.JKL`. (Observe que ele **não** se aplica a `ABC.JKL`; * usado nesse contexto sempre indica um qualificador.)
- Um caractere dentro de um qualificador em um nome de perfil para corresponder a zero ou mais caracteres dentro do qualificador em um nome de objeto.

  Por exemplo, `ABC.DE*.JKL` aplica-se aos objetos `ABC.DE.JKL`, `ABC.DEF.JKL` e `ABC.DEGH.JKL`.

******

Use o asterisco duplo (**) **uma vez** em um nome de perfil como:

- O nome do perfil inteiro para corresponder a todos os nomes de objeto. Por exemplo, se você usar `-t prcs` para identificar processos e, em seguida, usar ** como o nome do perfil, irá alterar as autorizações de todos os processos.
- Como o qualificador inicial, do meio ou final em um nome de perfil para corresponder a zero ou mais qualificadores em um nome de objeto. Por exemplo, `**.ABC` identifica todos os objetos com o qualificador final ABC.

Nota: Ao usar caracteres curinga em sistemas UNIX and Linux **deve-se** colocar o nome de perfil entre aspas simples.

Prioridades do Perfil

Mais de um perfil genérico pode aplicar-se a um único objeto. Quando for esse o caso, a regra mais específica se aplicará.

Um ponto importante a ser entendido ao usar perfis genéricos é a prioridade que os perfis recebem ao decidirem quais autoridades aplicar a um objeto que está sendo criado. Por exemplo, suponha que você emitiu estes comandos:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

O primeiro fornece autoridade put para todas as filas para o fred principal com nomes que correspondem ao perfil AB. *; O segundo fornece autoridade get para os mesmos tipos de fila que correspondem ao perfil AB.C*.

Suponha que agora você crie uma fila chamada AB.CD. De acordo com as regras para correspondência de curinga, setmqaut poderia ser aplicado a essa fila. Portanto, ele tem autoridade put ou get?

Para localizar a resposta, você aplica a regra que, sempre que vários perfis puderem se aplicar a um objeto, **apenas o mais específico será aplicado**. A maneira de aplicação dessa regra é comparando-se os nomes dos perfis da esquerda para a direita. Sempre que forem diferentes, um caractere não genérico

será mais específico do que um genérico. Portanto, no exemplo acima, a fila AB.CD tem autoridade **get** (AB.C* é mais específico que AB. *).

Ao comparar os caracteres genéricos, a ordem de *especificidade* é:

1. ?
2. *
3. **

Fazendo Dump de Configurações do Perfil

Use o comando de controle **dmpmqaut** ou o comando PCF **MQCMD_INQUIRE_AUTH_RECS** para fazer dump das autorizações atuais associadas a um perfil especificado

Para obter uma definição completa do comando de controle **dmpmqaut** e sua sintaxe, consulte dmpmqaute, para obter uma definição completa do comando PCF **MQCMD_INQUIRE_AUTH_RECS** e sua sintaxe, consulte Consultar Registros de Autoridade.

Os seguintes exemplos mostram o uso do comando de controle **dmpmqaut** para fazer dump de registros de autoridade de perfis genéricos:

1. Este exemplo faz dump de todos os registros de autoridade com um perfil que corresponde à fila a.b.c do principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   O dump resultante é semelhante a este exemplo:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Nota: Os usuários UNIX and Linux não podem usar a opção `-p` ; eles devem usar `-g groupname` no lugar.

2. Este exemplo faz dump de todos os registros de autoridade com um perfil que corresponde à fila a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   O dump resultante é semelhante a este exemplo:

   ```
   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. Este exemplo faz dump de todos os registros de autoridade para o perfil a.b. *, de fila do tipo

   ```
   dmpmqaut -m qmgr1 -n a.b.* -t q
   ```

   O dump resultante é semelhante a este exemplo:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   ```

```
type:        principal
authority:   get, browse, put, inq
```

4. Este exemplo faz dump de todos os registros de autoridade para o gerenciador de filas qmX.

   ```
   dmpmqaut -m qmX
   ```

   O dump resultante é semelhante a este exemplo:

   ```
   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. Este exemplo faz dump de todos os nomes de perfis e tipos de objetos para o gerenciador de filas qmX.

   ```
   dmpmqaut -m qmX -l
   ```

   O dump resultante é semelhante a este exemplo:

   ```
   profile: q1, type: queue
   profile: q*, type: queue
   profile: name.*, type: namelist
   profile: pr1, type: process
   ```

Nota: Para WebSphere MQ para Windows apenas, todos os proprietários exibidos incluem informações de domínio, por exemplo:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

Exibindo Configurações de Acesso

Use o comando de controle **dspmqaut** ou o comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** para visualizar as autorizações que um proprietário ou grupo específico possui para um objeto específico

O gerenciador de filas deve estar em execução para usar esse comando. Ao mudar o acesso de um diretor, as mudanças são refletidas imediatamente pelo OAM. A autorização pode ser exibida para apenas um grupo ou diretor de cada vez. Para obter uma definição completa do comando de controle **dmpmqaut** e sua sintaxe, consulte dmpmqaute para obter uma definição completa do comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** e sua sintaxe, consulte Inquire Entity Authority.

O exemplo a seguir mostra o uso do comando de controle **dspmqaut** para exibir as autorizações que o grupo `GpAdmin` tem para uma definição de processo chamada `Annuities` que está no gerenciador de filas `QueueMan1`

```
dspmqaut -m QueueMan1 -t process -n Annuities -g GpAdmin
```

Alterando e revogando o acesso a um objeto do IBM WebSphere MQ

Para alterar o nível de acesso que um usuário ou grupo tem para um objeto, utilize o comando **setmqaut** Para revogar o acesso de um usuário específico que é um membro de um grupo que possui autorização, remova o usuário do grupo

O processo de remover o usuário de um grupo é descrito em:

- “Criando e Gerenciando Grupos no Windows” na página 84
- “Criando e Gerenciando Grupos no HP-UX” na página 86
- “Criando e Gerenciando Grupos no AIX” na página 88
- “Criando e Gerenciando Grupos no Solaris” na página 89
- “Criando e gerenciando grupos no Linux” na página 90

.

O ID do usuário que cria um objeto do IBM WebSphere MQ é concedido autoridades de controle totais para esse objeto. Se você remover esse ID do usuário do grupo mqm local (ou do grupo Administradores em sistemas Windows), essas autoridades não serão revogadas. Use o comando de controle **setmqaut** ou o comando PCF **MQCMD_DELETE_AUTH_REC** para revogar o acesso a um objeto para o ID do usuário que o criou, depois de removê-lo do grupo mqm ou do grupo de Administradores. Para obter uma definição completa do comando de controle setmqaut e sua sintaxe, consulte setmqaute para obter uma definição completa do comando PCF **MQCMD_INQUIRE_ENTITY_AUTH** e sua sintaxe, consulte Inquire Entity Authority.

No Windows, exclua as entradas do OAM correspondentes a uma conta do usuário específica do Windows antes de excluir o perfil do usuário É impossível remover as entradas OAM após remover a conta do usuário.

Evitando verificações de acesso de segurança nos sistemas UNIX, Linux, and Windows

Para desativar toda a verificação de segurança, é possível desativar o OAM. Isso pode ser adequado para um ambiente de teste. Depois de desativar ou remover o OAM, não é possível incluir um OAM em um gerenciador de filas existente.

Se você decidir que não deseja executar verificações de segurança (por exemplo, em um ambiente de teste), poderá desativar o OAM em uma de duas maneiras:

- Antes de criar um gerenciador de filas, configure a variável de ambiente do sistema operacional MQSNOAUT (se você fizer isso, não será possível incluir um OAM posteriormente):

  Consulte Variáveis de Ambiente para obter mais informações sobre as implicações da configuração da variável MQSNOAUT.
- Edite o arquivo de configuração do gerenciador de filas para remover o serviço. (Se você fizer isso, não será possível incluir um OAM mais tarde.)

Se você usar setmqaut ou dspmqaut enquanto o OAM está desativado, observe os seguintes pontos:

- O OAM não valida o principal ou o grupo especificado, o que significa que o comando pode aceitar valores inválidos
- O OAM não executa verificações de segurança e indica que todos os principais e grupos têm autorização para executar todas as operações de objeto aplicáveis.

Aviso: Quando um OAM é removido, ele não pode ser colocado de volta em um gerenciador de filas existente. Isso ocorre porque o OAM precisa estar no local no momento da criação do objeto Para usar o OAM do WebSphere MQ novamente após ele ter sido removido, o gerenciador de filas precisa ser reconstruído

Conceitos relacionados

Serviços instaláveis

Concedendo acesso necessário para recursos

Use este tópico para determinar quais tarefas executar para aplicar segurança ao seu sistema WebSphere MQ ..

Sobre esta tarefa

Durante esta tarefa, você decide quais ações são necessárias para aplicar o nível apropriado de segurança aos elementos da instalação do WebSphere MQ . Cada tarefa individual para a qual você é encaminhado fornece instruções passo a passo para todas as plataformas.

Procedimento

1. Você precisa limitar o acesso ao gerenciador de filas para determinados usuários?
   a) Não: não executar ação adicional.
   b) Sim: Acesse a próxima pergunta.
2. Esses usuários precisam de acesso administrativo parcial em um subconjunto de recursos de gerenciadores de filas?
   a) Não: Acesse a próxima pergunta.
   b) Sim: Consulte “Concedendo Acesso Administrativo Parcial em um Subconjunto de Recursos do Gerenciador de Filas” na página 171.
3. Esses usuários precisam de acesso administrativo total em um subconjunto de recursos de gerenciadores de filas?
   a) Não: Acesse a próxima pergunta.
   b) Sim: Consulte “Concedendo Acesso Administrativo Total em um Subconjunto de Recursos do Gerenciador de Filas” na página 176.
4. Esses usuários precisam de acesso somente leitura a todos os recursos de gerenciadores de filas?
   a) Não: Acesse a próxima pergunta.
   b) Sim: Consulte “Concedendo Acesso somente Leitura a todos os Recursos em um Gerenciador de Filas” na página 181.
5. Esses usuários precisam de acesso administrativo total em todos recursos de gerenciadores de filas?
   a) Não: Acesse a próxima pergunta.
   b) Sim: Consulte “Concedendo Acesso Administrativo Total a todos os Recursos em um Gerenciador de Filas” na página 182.
6. Você precisa que os aplicativos de usuários se conectem ao gerenciador de filas?
   a) Não: Desative a conectividade, conforme descrito em “Removendo a Conectividade com o Gerenciador de Filas” na página 183
   b) Sim: Consulte “Permitindo que Aplicativos de Usuário se Conectem ao Gerenciador de Filas” na página 184.

Concedendo Acesso Administrativo Parcial em um Subconjunto de Recursos do Gerenciador de Filas

É necessário fornecer a determinados usuários acesso administrativo parcial a alguns recursos do gerenciador de filas, mas não a todos. Use esta tabela para determinar as ações que precisam ser executadas.

Tabela 14. Concedendo acesso administrativo parcial para um subconjunto de recursos do gerenciador de filas	
Os usuários precisam administrar objetos deste tipo	**Executar esta ação**
Filas	Conceda acesso administrativo parcial às filas necessárias, conforme descrito em "Concedendo Acesso Administrativo Limitado a algumas Filas" na página 172
tópicos	Conceda acesso administrativo parcial aos tópicos necessários, conforme descrito em "Concedendo Acesso Administrativo Limitado a alguns Tópicos" na página 173
Canais	Conceda acesso administrativo parcial aos canais necessários, conforme descrito em "Concedendo Acesso Administrativo Limitado a alguns Canais" na página 173
O gerenciador de filas	Conceda acesso administrativo parcial ao gerenciador de filas, conforme descrito em "Concedendo Acesso Administrativo Limitado a um Gerenciador de Filas" na página 174
Processos	Conceda acesso administrativo parcial aos processos necessários, conforme descrito em "Concedendo Acesso Administrativo Limitado a alguns Processos" na página 175
Listas de Nomes	Conceda acesso administrativo parcial às listas de nomes necessárias, conforme descrito em "Concedendo Acesso Administrativo Limitado a algumas Listas de Nomes" na página 175
Serviços	Conceda acesso administrativo parcial aos serviços necessários, conforme descrito em "Concedendo Acesso Administrativo Limitado a alguns Serviços" na página 176

Concedendo Acesso Administrativo Limitado a algumas Filas

Conceda acesso administrativo parcial a algumas filas em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a algumas filas para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName ReqdAction
```

- Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

ReqdAction
A ação que você está permitindo que o grupo execute:

– Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + dlt, + dsp. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Nota: Conceder +crt para filas torna indiretamente o usuário ou o grupo um administrador. Não use a autoridade +crt para conceder acesso administrativo limitado a algumas filas.

QType

Para o comando DISPLAY, um dos valores QUEUE, QLOCAL, QALIAS, QMODEL, QREMOTE ou QCLUSTER.

Para outros valores de *ReqdAction*, um dos valores QLOCAL, QALIAS, QMODEL ou QREMOTE.

Concedendo Acesso Administrativo Limitado a alguns Tópicos

Conceda acesso administrativo parcial a alguns tópicos em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a alguns tópicos para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName ReqdAction
```

- Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

ReqdAction
A ação que você está permitindo que o grupo execute:

– Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + dsp. + ctrl. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Concedendo Acesso Administrativo Limitado a alguns Canais

Conceda acesso administrativo parcial a alguns canais em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a alguns canais para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName ReqdAction
```

- Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

  ReqdAction
  A ação que você está permitindo que o grupo execute:

  – Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + dsp. + ctrl, + ctrlx. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Concedendo Acesso Administrativo Limitado a um Gerenciador de Filas

Conceda acesso administrativo parcial a um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado para executar algumas ações no gerenciador de filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName ReqdAction
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*MQM) USER(GroupName) AUT(ReqdAction)
MQMNAME('QMgrName')
```

Resultados

Para determinar quais comandos MQSC o usuário pode executar no gerenciador de filas, emita os seguintes comandos para cada comando MQSC:

```
RDEFINE MQCMDS QMgrName.ReqdAction.QMGR UACC(NONE)
PERMIT QMgrName.ReqdAction.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Para permitir que o usuário use o comando DISPLAY QMGR, emita os seguintes comandos:

```
RDEFINE MQCMDS QMgrName.DISPLAY.QMGR UACC(NONE)
PERMIT QMgrName.DISPLAY.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

ReqdAction
A ação que você está permitindo que o grupo execute:

- Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + dsp. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

  Embora +set seja uma autorização do MQI e não considerada administrativa normalmente, conceder +set no gerenciador de filas pode levar indiretamente à autoridade administrativa total. Não conceda +set a usuários e aplicativos ordinários.

Concedendo Acesso Administrativo Limitado a alguns Processos

Conceda acesso administrativo parcial a alguns processos em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a alguns processos para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName ReqdAction
  ```

- Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

  ReqdAction
  A ação que você está permitindo que o grupo execute:

  – Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + dsp. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Concedendo Acesso Administrativo Limitado a algumas Listas de Nomes

Conceda acesso administrativo parcial a algumas listas de nomes em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a algumas listas de nomes para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName ReqdAction
  ```

- Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

ReqdAction
A ação que você está permitindo que o grupo execute:

– Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Concedendo Acesso Administrativo Limitado a alguns Serviços

Conceda acesso administrativo parcial a alguns serviços em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo limitado a alguns serviços para algumas ações, use os comandos apropriados de seu sistema operacional.

Nota: Objetos de serviço não existem no z/OS.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName ReqdAction
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*SVC) USER(GroupName) AUT(ReqdAction)
MQMNAME('QMgrName')
```

Resultados

Esses comandos concedem acesso ao serviço especificado. Para determinar quais comandos MQSC o usuário pode executar no serviço, emita os seguintes comandos para cada comando MQSC:

```
RDEFINE MQCMDS QMgrName.ReqdAction.SERVICE UACC(NONE)
PERMIT QMgrName.ReqdAction.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Para permitir que o usuário use o comando DISPLAY SERVICE, emita os seguintes comandos:

```
RDEFINE MQCMDS QMgrName.DISPLAY.SERVICE UACC(NONE)
PERMIT QMgrName.DISPLAY.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

ReqdAction
A ação que você está permitindo que o grupo execute:

- Nos sistemas UNIX, Linux e Windows , qualquer combinação das seguintes autorizações: + chg, + clr, + crt, + dlt, + ctrl, + ctrlx, + dsp. A autorização +alladm é equivalente a +chg +clr +dlt +dsp.

Concedendo Acesso Administrativo Total em um Subconjunto de Recursos do Gerenciador de Filas

É necessário fornecer a determinados usuários acesso administrativo total a alguns recursos do gerenciador de filas, mas não a todos. Use estas tabelas para determinar as ações que precisam ser executadas.

Tabela 15. Concedendo acesso administrativo completo para um subconjunto de recursos do gerenciador de filas

Os usuários precisam administrar objetos deste tipo	**Executar esta ação**
Filas	Conceda acesso administrativo total às filas necessárias, conforme descrito em "Concedendo Acesso Administrativo Total a algumas Filas" na página 177
tópicos	Conceda acesso administrativo total aos tópicos necessários, conforme descrito em "Concedendo Acesso Administrativo Total a alguns Tópicos" na página 178
Canais	Conceda acesso administrativo total aos canais necessários, conforme descrito em "Concedendo Acesso Administrativo Total a alguns Canais" na página 178
O gerenciador de filas	Conceda acesso administrativo total ao gerenciador de filas, conforme descrito em "Concedendo Acesso Administrativo Total a um Gerenciador de Filas" na página 179
Processos	Conceda acesso administrativo total aos processos necessários, conforme descrito em "Concedendo Acesso Administrativo Total a alguns Processos" na página 179
Listas de Nomes	Conceda acesso administrativo total às listas de nomes necessárias, conforme descrito em "Concedendo Acesso Administrativo Total a algumas Listas de Nomes" na página 180
Serviços	Conceda acesso administrativo total aos serviços necessários, conforme descrito em "Concedendo Acesso Administrativo Total a alguns Serviços" na página 181

Concedendo Acesso Administrativo Total a algumas Filas

Conceda acesso administrativo total a algumas filas em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a algumas filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.QUEUE.ObjectProfile UACC(NONE)
PERMIT QMgrName.QUEUE.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a alguns Tópicos

Conceda acesso administrativo total a alguns tópicos em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a alguns tópicos para algumas ações, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*TOPIC) USER(GroupName) AUT(ALLADM)
MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.TOPIC.ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a alguns Canais

Conceda acesso administrativo total a alguns canais em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a alguns canais, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*CHL) USER(GroupName) AUT(ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.CHANNEL.ObjectProfile UACC(NONE)
PERMIT QMgrName.CHANNEL.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a um Gerenciador de Filas

Conceda acesso administrativo total a um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total ao gerenciador de filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -t qmgr -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*MQM) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.QMGR UACC(NONE)
PERMIT QMgrName.QMGR CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a alguns Processos

Conceda acesso administrativo total a alguns processos em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a alguns processos, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*PRC) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.CHANNEL.ObjectProfile UACC(NONE)
PERMIT QMgrName.PROCESS.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a algumas Listas de Nomes

Conceda acesso administrativo total a algumas listas de nomes em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a algumas listas de nomes, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALLADM)
MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.NAMELIST.ObjectProfile UACC(NONE)
PERMIT QMgrName.NAMELIST.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a alguns Serviços

Conceda acesso administrativo total a alguns serviços em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder acesso administrativo total a alguns serviços, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName +alladm
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*SVC) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.SERVICE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SERVICE.ObjectProfile CLASS(MQADMIN) ID(GroupName ) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso somente Leitura a todos os Recursos em um Gerenciador de Filas

Conceda acesso somente leitura a todos os recursos em um gerenciador de filas, a cada usuário ou grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Use o assistente Incluir autoridades baseadas na função ou os comandos apropriados de seu sistema operacional.

Procedimento

- Usando o assistente:
  a) Na área de janela do WebSphere MQ Explorer Navigator , clique com o botão direito no gerenciador de filas e clique em **Autoridades baseadas em objeto** > **Incluir autoridades baseadas em função**

  O assistente Incluir Autoridades Baseadas na Função é aberto.
- Para sistemas UNIX e Windows , emita os comandos a seguir:

```
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
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
```

As autoridades específicas para o SYSTEM.ADMIN.COMMAND.QUEUE e SYSTEM.MQEXPLORER.REPLY.MODEL são necessários apenas se você desejar utilizar o MQ Explorer

- Para IBM i, emita os comandos a seguir:

```
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')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQQUEUE QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQQUEUE) ID(GroupName) ACCESS(READ)
RDEFINE MQTOPIC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQTOPIC) 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)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

GroupName
O nome do grupo que receberá acesso.

Concedendo Acesso Administrativo Total a todos os Recursos em um Gerenciador de Filas

Conceda acesso administrativo total a todos os recursos em um gerenciador de filas, a cada usuário ou grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Use o assistente Incluir autoridades baseadas na função ou os comandos apropriados de seu sistema operacional.

Procedimento

- Usando o assistente:
  a) Na área de janela do WebSphere MQ Explorer Navigator , clique com o botão direito no gerenciador de filas e clique em **Autoridades baseadas em objeto** > **Incluir autoridades baseadas em função**
  O assistente Incluir Autoridades Baseadas na Função é aberto.

- Para sistemas UNIX and Linux, emita os comandos a seguir:

```
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.QUEUE -t queue -g GroupName +dsp +inq +get
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 +conn
```

- Para sistemas Windows, emita os mesmos comandos que para sistemas UNIX and Linux , mas usando o nome do perfil @CLASS em vez de @class
- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*ALL) USER('GroupName') AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQADMIN QMgrName.*.** UACC(NONE)
PERMIT QMgrName.*.** CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

GroupName
O nome do grupo que receberá acesso.

Removendo a Conectividade com o Gerenciador de Filas

Para que aplicativos de usuário não se conectem ao gerenciador de filas, remova sua autoridade de conexão com ele.

Sobre esta tarefa

Revogue a autoridade de todos os usuários de conexão com o gerenciador de filas usando o comando apropriado de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -t qmgr -g GroupName -connect
```

- Para IBM i, emita o seguinte comando:

```
RVKMQMAUT OBJ ('QMgrName') OBJTYPE(*MQM) USER(*ALL) AUT(*CONNECT)
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
```

Não emita comandos PERMIT.

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

GroupName
O nome do grupo cujo acesso será negado.

Permitindo que Aplicativos de Usuário se Conectem ao Gerenciador de Filas

Você deseja permitir que o aplicativo de usuário se conecte ao gerenciador de filas. Use as tabelas deste tópico para determinar quais ações executar.

Primeiro, determine se os aplicativos clientes se conectarão ao seu Gerenciador de Filas.

Se nenhum dos aplicativos que irão se conectar ao gerenciador de filas for aplicativo cliente, desative o acesso remoto, conforme descrito em "Desativando o Acesso Remoto ao Gerenciador de Filas" na página 191.

Se um ou mais dos aplicativos que irão se conectar ao gerenciador de filas forem aplicativos clientes, proteja a conectividade remota, conforme descrito em "Protegendo a Conectividade Remota no Gerenciador de Filas" na página 184.

Em ambos os casos, configure a segurança de conexão, conforme descrito em "Configurando a Segurança de Conexão" na página 191

Para controlar o acesso a recursos de cada usuário que se conectar ao gerenciador de filas, consulte a tabela a seguir. Se a instrução na primeira coluna for verdadeira, execute a ação listada na segunda coluna.

Instrução	Execute esta ação
Você tem aplicativos que usam filas	Consulte "Controlando o Acesso de Usuário a Filas" na página 192
Você tem aplicativos que usam tópicos	Consulte o "Controlando o Acesso de Usuário aos Tópicos" na página 197.
Você tem aplicativos que consultam o objeto de gerenciador de filas	Consulte o "Concedendo autoridade para consultar em um gerenciador de filas" na página 198.
Você tem aplicativos que usam objetos de processos	Consulte "Concedendo autoridade para acessar processos" na página 199
Você tem aplicativos que usam listas de nomes	Consulte "Concedendo autoridade para acessar listas de nomes" na página 199

Protegendo a Conectividade Remota no Gerenciador de Filas

É possível proteger a conectividade remota com o gerenciador de filas usando SSL ou TLS, uma saída de segurança, registros de autenticação de canal ou uma combinação destes métodos.

Sobre esta tarefa

Você conecta um cliente ao gerenciador de filas usando um canal de conexão do cliente na estação de trabalho do cliente e um canal de conexão do servidor no servidor. Proteja essas conexões de uma das seguintes maneiras.

Procedimento

1. Usando SSL ou TLS com registros de autenticação de canal:

a) Evite que qualquer nome distinto (DN) abra um canal, usando um registro de autenticação de canal SSLPEERMAP para mapear todos os DNs para USERSRC(NOACCESS).

b) Permita que DNs específicos ou conjuntos de DNs abram um canal, usando um registro de autenticação de canal SSLPEERMAP para mapeá-los para USERSRC(CHANNEL).

2. Usando SSL ou TLS com uma saída de segurança:
   a) Configure MCAUSER no canal de conexão do servidor para um identificador de usuários sem privilégios.
   b) Grave uma saída de segurança para designar um valor MCAUSER, dependendo do valor do DN SSL que ele recebe nos campos SSLPeerNamePtr e SSLPeerNameLength transmitidos para a saída na estrutura MQCD.
3. Usando SSL ou TLS com valores de definição de canal fixos:
   a) Configure SSLPEER no canal de conexão do servidor para um valor específico ou limite o intervalo de valores.
   b) Configure MCAUSER no canal de conexão do servidor para o ID do usuário com o qual o canal deve ser executado.
4. Usando registros de autenticação de canal em canais que não usam SSL ou TLS:
   a) Evite que qualquer endereço IP abra canais, usando um registro de autenticação de canal de mapeamento de endereço com ADDRESS(*) e USERSRC(NOACCESS).
   b) Permita que endereços IP específicos abram canais, usando registros de autenticação de canal de mapeamento de endereço para esses endereços com USERSRC(CHANNEL).
5. Usando uma saída de segurança:
   a) Grave uma saída de segurança para autorizar conexões com base em qualquer propriedade escolhida, por exemplo, o endereço IP de origem.
6. Também é possível usar registros de autenticação de canal com uma saída de segurança ou usar todos os três métodos, se suas circunstâncias específicas exigirem.

Bloqueando Endereços IP Específicos
É possível evitar que um canal específico aceite uma conexão de entrada de um endereço IP, ou evitar que o gerenciador de filas inteiro permita o acesso a partir de um endereço IP, usando um registro de autenticação de canal.

Antes de começar

Ative registros de autenticação de canal executando o comando a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Sobre esta tarefa

Para desaprovar canais específicos para aceitação de uma conexão de entrada e assegurar que conexões sejam aceitas apenas ao usar o nome de canal correto, um tipo de regra pode ser usado para bloquear endereços IP. Para desaprovar um endereço IP a acessar o gerenciador de fias inteiro, você normalmente usaria um firewall para bloqueá-lo permanentemente. Entretanto, um outro tipo de regra pode ser usado para permitir que você bloqueie alguns endereços temporariamente, por exemplo, enquanto você estiver aguardando pela atualização do firewall.

Procedimento

- Para bloquear o uso de endereços IP, configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**ou o comando PCF **Set Channel Authentication Record**.

```
SET CHLAUTH(generic-channel-name) TYPE(ADDRESSMAP) ADDRESS(generic-ip-address)
USERSRC(NOACCESS)
```

Existem três partes para o comando:

SET CHLAUTH (*generic-channel-name*)
Use esta parte do comando para controlar se você deseja bloquear uma conexão para o gerenciador de filas inteiro, canal único ou intervalo de canais. O que você colocou aqui determina quais áreas estão cobertas.

Por exemplo:

- SET CHLAUTH('*') - bloqueia todos os canais em um gerenciador de filas, ou seja, todo o gerenciador de filas
- SET CHLAUTH('SYSTEM.*') - bloqueia todos os canais iniciados com SYSTEM.
- SET CHLAUTH('SYSTEM.DEF.SVRCONN') - bloqueia o canal SYSTEM.DEF.SVRCONN

Tipo de regra CHLAUTH
Use esta parte do comando para especificar o tipo de comando e determinar se você deseja fornecer um único endereço ou uma lista de endereços.

Por exemplo:

- TYPE(ADDRESSMAP) - Use ADDRESSMAP se você deseja fornecer um único endereço ou um endereço curinga. Por exemplo, ADDRESS('192.168.*') bloqueia quaisquer conexões provenientes de um endereço IP com início no 192.168.

  Para obter mais informações sobre como filtrar endereços IP com padrões, consulte Endereços IP genéricos.
- TYPE(BLOCKADDR) - Use BLOCKADDR se você deseja fornecer uma lista de endereços para o bloqueio.

Parâmetros Adicionais
Esses parâmetros são dependentes do tipo de regra usada na segunda parte do comando:

- Para TYPE(ADDRESSMAP), use o ADDRESS
- Para TYPE(BLOCKADDR), use o ADDRLIST

Referências relacionadas
SET CHLAUTH

Bloqueando temporariamente endereços IP específicos se o gerenciador de filas não estiver em execução
Você pode desejar bloquear determinados endereços IP ou intervalos de endereços, quando o gerenciador de filas não estiver em execução e não é possível, portanto, emitir comandos MQSC. É possível bloquear temporariamente os endereços IP em uma base excepcional modificando o arquivo `blockaddr.ini`.

Sobre esta tarefa

O arquivo `blockaddr.ini` contém uma cópia das definições de BLOCKADDR que são usadas pelo Gerenciador de Filas. Esse arquivo é lido pelo listener se o listener é iniciado antes do gerenciador de filas. Nessas circunstâncias, o listener utiliza quaisquer valores que você tenha incluído manualmente no arquivo `blockaddr.ini`

No entanto, esteja ciente de que quando o Gerenciador de Filas é iniciado, ele grava o conjunto de definições de BLOCKADDR no arquivo `blockaddr.ini`, substituindo qualquer edição manual que você possa ter feito. Da mesma forma, sempre que você incluir ou excluir uma definição de BLOCKADDR usando o comando **SET CHLAUTH**, o arquivo `blockaddr.ini` é atualizado. É possível, portanto, tornar as mudanças permanentes para as definições de BLOCKADDR somente usando o comando **SET CHLAUTH** quando o gerenciador de filas estiver em execução.

Procedimento

1. Abra o arquivo `blockaddr.ini` em um editor de texto.
   O arquivo está localizado no diretório de dados do gerenciador de filas.
2. Inclua endereços IP como pares de valor de palavra-chave simples, em que a palavra-chave é `Addr`.

Para obter informações sobre filtrar endereços IP com padrões, consulte Endereços IP genéricos.

Por exemplo:

```
Addr = 192.0.2.0
Addr = 192.0.*
Addr = 192.0.2.1-8
```

Tarefas relacionadas

“Bloqueando Endereços IP Específicos” na página 185

É possível evitar que um canal específico aceite uma conexão de entrada de um endereço IP, ou evitar que o gerenciador de filas inteiro permita o acesso a partir de um endereço IP, usando um registro de autenticação de canal.

Referências relacionadas

SET CHLAUTH

Bloqueando IDs de Usuários Específicos

É possível evitar que usuários específicos usem um canal especificando IDs de usuários que, se declarados, fazem com que o canal termine. Faça isso configurando um registro de autenticação de canal de canal.

Antes de começar

Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimento

Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(BLOCKUSER) USERLIST(userID1, userID2)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.

A lista de usuários fornecida em um TYPE(BLOCKUSER) se aplica somente a canais SVRCONN e não a canais de gerenciador de filas para gerenciador de filas.

userID1 e *userID2* são cada um o ID de um usuário que deve ser evitado de usar o canal. Também é possível especificar o valor especial *MQADMIN para fazer referência a usuários administrativos privilegiados. Para obter informações adicionais sobre usuários privilegiados, consulte “Usuários Privilegiados” na página 149. Para obter informações adicionais sobre *MQADMIN, consulte SET CHLAUTH.

Referências relacionadas

SET CHLAUTH

Mapeando um Gerenciador de Filas Remoto para um ID do Usuário MCAUSER

É possível usar um registro de autenticação de canal para configurar o atributo MCAUSER de um canal de acordo com o gerenciador de filas a partir do qual o canal está conectando.

Antes de começar

Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Sobre esta tarefa

Como opção, é possível restringir os endereços IP aos quais a regra se aplica.

Observe que essa técnica não se aplica a canais de conexão do servidor. Se você especificar o nome de um canal de conexão do servidor nos comandos mostrados abaixo, ele não terá efeito.

Procedimento

- Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
) USERSRC(MAP) MCAUSER(user)
```

  generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
  generic-partner-qmgr-name é o nome do gerenciador de filas ou um padrão que inclui o símbolo de asterisco (*) como um curinga que corresponde ao nome do gerenciador de filas.
  user é o ID do usuário a ser usado para todas as conexões do gerenciador de filas especificado.

- Para restringir esse comando a determinados endereços IP, inclua o parâmetro **ADDRESS** da seguinte forma:

```
SET CHLAUTH('generic-channel-name') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
) USERSRC(MAP) MCAUSER(user) ADDRESS(
generic-ip-address)
```

  generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
  generic-ip-address é um endereço único ou um padrão que inclui o símbolo de asterisco (*) como um curinga ou o hífen (-) para indicar um intervalo, que corresponda ao endereço. Para obter mais informações sobre endereços IP genéricos, consulte Endereços IP genérico

Referências relacionadas
SET CHLAUTH

Mapeando um ID do Usuário Decrarado pelo Cliente para um ID do Usuário MCAUSER
É possível usar um registro de autenticação de canal para alterar o atributo MCAUSER de um canal de conexão do servidor, de acordo com o ID do usuário original recebido de um cliente.

Antes de começar
Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Sobre esta tarefa
Observe que essa técnica se aplica somente a canais de conexão do servidor. Não tem nenhum efeito em outros tipos de canais.

Procedimento

Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**ou o comando PCF **Set Channel Authentication Record** . Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (USERMAP) CLNTUSER(client-user-name) USERSRC(MAP)
MCAUSER(
user)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.

client-user-name é o ID do usuário declarado pelo cliente.
user é o ID do usuário a ser usado em vez de o nome de usuário do cliente.

Referências relacionadas
SET CHLAUTH

Mapeando um Nome Distinto de SSL ou TLS para um ID do Usuário MCAUSER
É possível usar um registro de autenticação de canal para configurar o atributo MCAUSER de um canal de acordo com o Nome Distinto (DN) recebido.

Antes de começar
Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimento
Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (SSLPEERMAP) SSLPEER(generic-ssl-peer-name
) USERSRC(MAP) MCAUSER(user)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
generic-ssl-peer-name é uma d que segue as regras de padrão do IBM WebSphere MQ para valores de SSLPEER. Consulte as regras do WebSphere MQ para obter valores SSLPEER.
user é o ID do usuário a ser usado para todas as conexões usando o DN especificado.

Referências relacionadas
SET CHLAUTH

Bloqueando Acesso de um Gerenciador de Filas Remotas
É possível usar um registro de autenticação de canal para evitar que um gerenciador de filas remotas inicie canais.

Antes de começar
Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Sobre esta tarefa
Observe que essa técnica não se aplica a canais de conexão do servidor. Se você especificar o nome de um canal de conexão do servidor no comando mostrado abaixo, ele não terá efeito.

Procedimento
Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(QMGRMAP) QMNAME('generic-partner-qmgr-name')
USERSRC(NOACCESS)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
generic-partner-qmgr-name é o nome do gerenciador de filas ou um padrão que inclui o símbolo de asterisco (*) como um curinga que corresponde ao nome do gerenciador de filas.

Referências relacionadas
SET CHLAUTH

Bloqueando Acesso para um ID do Usuário Declarado do Cliente
É possível usar um registro de autenticação de canal para evitar que um ID do usuário declarado do cliente inicie canais.

Antes de começar
Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Sobre esta tarefa
Observe que essa técnica se aplica somente a canais de conexão do servidor. Não tem nenhum efeito em outros tipos de canais.

Procedimento
Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(USERMAP) CLNTUSER('client-user-name') USERSRC(NOACCESS)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
client-user-name é o ID do usuário declarado pelo cliente.

Referências relacionadas
SET CHLAUTH

Bloqueando Acesso para um Nome Distinto de SSL
É possível usar um registro de autenticação de canal para evitar que um Nome Distinto SSL inicie canais.

Antes de começar
Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimento
Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(SSLPEERMAP) SSLPEER('generic-ssl-peer-name')
USERSRC(NOACCESS)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.
generic-ssl-peer-name é uma d que segue as regras de padrão do IBM WebSphere MQ para valores de SSLPEER. Consulte as regras do WebSphere MQ para obter valores SSLPEER.

Referências relacionadas
SET CHLAUTH

Mapeando um Endereço IP para um ID do Usuário MCAUSER
É possível usar um registro de autenticação de canal para configurar o atributo MCAUSER de um canal de acordo com o endereço IP a partir do qual a conexão é recebida.

Antes de começar

Certifique-se de que os registros de autenticação de canal são ativados como a seguir:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimento

Configure um registro de autenticação de canal usando o comando MQSC **SET CHLAUTH**, ou o comando PCF **Set Channel Authentication Record**. Por exemplo, é possível emitir o comando de MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(ADDRESSMAP) ADDRESS('generic-ip-address') USERSRC(MAP)
MCAUSER(user)
```

generic-channel-name é o nome de um canal para o qual você deseja acesso ao controle, ou um padrão incluindo o símbolo asterisco (*) como um curinga que corresponde ao nome de canal.

user é o ID do usuário a ser usado para todas as conexões usando o DN especificado.

generic-ip-address é o endereço a partir do qual a conexão está sendo feita ou um padrão que inclui o asterisco (*) como um curinga ou o hífen (-) para indicar um intervalo, que corresponda ao endereço.

Referências relacionadas

SET CHLAUTH

Desativando o Acesso Remoto ao Gerenciador de Filas

Para que os aplicativos clientes não sejam conectados ao gerenciador de filas, desative o acesso remoto a eles.

Sobre esta tarefa

Evite que aplicativos clientes sejam conectados ao gerenciador de filas em uma das seguintes maneiras:

Procedimento

- Exclua todos os canais de conexão do servidor usando o comando **DELETE CHANNEL**do MQSC
- Configure o identificador de usuários do agente do canal de mensagens (MCAUSER) do canal com um ID do usuário sem direitos de acesso, usando o comando MQSC **ALTER CHANNEL**.

Configurando a Segurança de Conexão

Conceda a autoridade de conexão com o gerenciador de filas a cada usuário ou grupo de usuários que tiver uma necessidade de negócios.

Sobre esta tarefa

Para configurar a segurança de conexão, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

  ```
  setmqaut -m QMgrName -t qmgr -g GroupName +connect
  ```

- Para IBM i, emita o seguinte comando:

  ```
  GRTMQMAUT OBJ('QMgrName') OBJTYPE(*MQM)  USER('GroupName') AUT(*CONNECT)
  ```

- Para z/OS, emita os comandos a seguir:

  ```
  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)
```

Esses comandos fornecem autoridade para se conectar para lote, CICS, IMS e o inicializador de canais (CHIN). Se você não usar um tipo particular de conexão, omita os comandos relevantes.

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Controlando o Acesso de Usuário a Filas

Você deseja controlar o acesso ao aplicativo a filas. Use este tópico para determinar quais ações executar.

Para cada instrução verdadeira na primeira coluna, execute a ação indicada na segunda coluna.

Instrução	Ação
O aplicativo obtém mensagens de uma fila	Consulte “Concedendo autoridade para obter mensagens de filas” na página 192
O aplicativo configura contexto	Consulte “Concedendo autoridade para configurar o contexto” na página 193
O aplicativo passa contexto	Consulte “Concedendo autoridade para passar o contexto” na página 194
O aplicativo coloca mensagens em uma fila armazenada em cluster	Consulte “Autorizando a Colocação de Mensagens em Filas de Cluster Remotas” na página 250
O aplicativo coloca mensagens em uma fila local	Consulte “Concedendo autoridade para colocar mensagens em uma fila local” na página 194
O aplicativo coloca mensagens em uma fila modelo	Consulte “Concedendo autoridade para colocar mensagens em uma fila modelo” na página 195
O aplicativo coloca mensagens em uma fila remota	Consulte “Concedendo autoridade para colocar mensagens em uma fila do cluster remoto” na página 196

Concedendo autoridade para obter mensagens de filas

Conceda a autoridade para obter mensagens de uma fila ou um conjunto de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para obter mensagens de algumas filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +get
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*GET) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para configurar o contexto
Conceda a autoridade para configurar contexto em uma mensagem que está sendo colocada, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para configurar contexto em algumas filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita um dos comandos a seguir:
  - Para configurar apenas contexto de identidade:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setid
```

  - Para configurar todo o contexto:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setall
```

- Para o IBM i, emita um dos comandos a seguir:
  - Para configurar apenas contexto de identidade:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*SETID) MQMNAME('QMgrName')
```

  - Para configurar todo o contexto:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*SETALL)
MQMNAME('QMgrName')
```

- Para z/OS, emita um dos seguintes conjuntos de comandos:
  - Para configurar apenas contexto de identidade:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  - Para configurar todo o contexto:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(CONTROL)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para passar o contexto
Conceda a autoridade para passar contexto de uma mensagem recuperada para uma que está sendo colocada, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para passar contexto em algumas filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita um dos comandos a seguir:
  - Para passar apenas contexto de identidade:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passid
    ```

  - Para passar todo o contexto:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passall
    ```

- Para o IBM i, emita um dos comandos a seguir:
  - Para passar apenas contexto de identidade:

    ```
    GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*PASSID)
    MQMNAME('QMgrName')
    ```

  - Para passar todo o contexto:

    ```
    GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*PASSALL)
    MQMNAME('QMgrName')
    ```

- Para z/OS, emita os comandos a seguir para transmitir o contexto de identidade ou todo o contexto:

  ```
  RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
  PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
  ```

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

Concedendo autoridade para colocar mensagens em uma fila local
Conceda a autoridade para colocar mensagens em uma fila local ou um conjunto de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para colocar mensagens em algumas filas locais, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

Concedendo autoridade para colocar mensagens em uma fila modelo
Conceda a autoridade para colocar mensagens em uma fila modelo ou um conjunto de filas modelo, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Filas modelo são usadas para criar filas dinâmicas. Você deve, portanto, conceder autoridade para ambas as filas, modelo e dinâmica. Para conceder essas autoridades, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita os comandos a seguir:

```
setmqaut -m QMgrName -n ModelQueueName -t queue -g GroupName +put
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Para IBM i, emita os comandos a seguir:

```
GRTMQMAUT OBJ('ModelQueueName') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME('QMgrName')
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
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)
```

  Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ModelQueueName
O nome da fila modelo em que as filas dinâmicas se baseiam.

ObjectProfile
O nome do perfil da fila dinâmica ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para colocar mensagens em uma fila do cluster remoto
Conceda a autoridade para colocar mensagens em uma fila cluster remoto ou um conjunto de filas a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para colocar uma mensagem em uma fila do cluster remoto, é possível colocá-la em uma definição local de uma fila remota ou uma fila remota completa. Se você estiver usando uma definição local de uma fila remota, você precisa de autoridade para colocar o objeto local: consulte “Concedendo autoridade para colocar mensagens em uma fila local” na página 194. Se você estiver usando uma fila remota completa, você precisa de autoridade para colocar a fila remota. Conceda esta autoridade usando os comandos apropriados para seu sistema operacional.

O comportamento padrão é realizar o controle de acesso no `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Observe que esse comportamento se aplica mesmo se você estiver usando diversas filas de transmissão.

O comportamento específico descritos neste tópico se aplica somente quando você tiver configurado o atributo **`ClusterQueueAccessControl`** no arquivo `qm.ini` para ser *RQMName*, conforme descrito no tópico Sub-rotina de segurança e reiniciou o gerenciador de filas.

Nos sistemas UNIX, Linuxe Windows , também é possível usar o comando SET AUTHREC

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -t rqmname -n
ObjectProfile -g GroupName +put
```

  Observe que é possível usar o objeto *rqmname* somente para filas de cluster remoto.

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJTYPE(*RMTMQMNAME) OBJ('
ObjectProfile') USER(GroupName) AUT(*PUT) MQMNAME('
QMgrName')
```

  Observe que é possível usar o objeto RMTMQMNAME somente para filas de cluster remoto.

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQQUEUE  QMgrNameObjectProfile UACC(NONE)
PERMIT QMgrNameObjectProfile CLASS(MQADMIN)
ID(GroupName) ACCESS(UPDATE)
```

  Observe que é possível usar o nome do gerenciador de filas remotas (ou grupo de compartilhamento de filas) somente para filas remotas do cluster.

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do gerenciador de filas remotas ou do perfil genérico para o qual mudar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Controlando o Acesso de Usuário aos Tópicos

É necessário controlar o acesso de aplicativos aos tópicos. Use este tópico para determinar quais ações executar.

Para cada instrução verdadeira na primeira coluna, execute a ação indicada na segunda coluna.

Tabela 16. Controlando o Acesso de Usuário aos Tópicos

Instrução	**Ação**
O aplicativo publica mensagens em um tópico	Consulte “Concedendo autoridade para publicar mensagens em um tópico” na página 197
O aplicativo assina um tópico	Consulte “Concedendo autoridade para assinar tópicos” na página 197

Concedendo autoridade para publicar mensagens em um tópico

Conceda a autoridade para publicar mensagens em um tópico ou um conjunto de tópicos, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para publicar mensagens em alguns tópicos, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +pub
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*TOPIC) USER(GroupName) AUT(*PUB) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQTOPIC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para assinar tópicos

Conceda a autoridade para assinar um tópico ou um conjunto de tópicos, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para assinar alguns tópicos, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +sub
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*TOPIC) USER(GroupName) AUT(*SUB) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQTOPIC QMgrName.SUBSCRIBE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SUBSCRIBE.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

  Os nomes das variáveis possuem os seguintes significados:

  QMgrName
  O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

  ObjectProfile
  O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

  GroupName
  O nome do grupo que receberá acesso.

Concedendo autoridade para consultar em um gerenciador de filas

Conceda a autoridade para consultar em um gerenciador de filas, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para consultar em um gerenciador de filas, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName +inq
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*MQM) USER(GroupName) AUT(*INQ) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQCMDS QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName ObjectProfile CLASS(MQCMDS) ID(GroupName ) ACCESS(READ)
```

  Esses comandos concedem acesso ao gerenciador de filas especificado. Para permitir que o usuário use o comando MQINQ, emita os seguintes comandos:

```
RDEFINE MQCMDS QMgrName.MQINQ.QMGR UACC(NONE)
PERMIT QMgrName.MQINQ.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

  Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para acessar processos

Conceda a autoridade para acessar um processo ou um conjunto de processos, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para acessar alguns processos, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +all
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*PRC) USER(GroupName) AUT(*ALL) MQMNAME('QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQPROC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQPROC) ID(GroupName) ACCESS(READ)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Concedendo autoridade para acessar listas de nomes

Conceda a autoridade para acessar uma lista de nomes ou um conjunto de listas de nomes, a cada grupo de usuários com uma necessidade de negócios para isso.

Sobre esta tarefa

Para conceder a autoridade para acessar algumas listas de nomes, use os comandos apropriados de seu sistema operacional.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita o comando a seguir:

```
setmqaut -m QMgrName -n
ObjectProfile -t namelist -g GroupName
 +all
```

- Para IBM i, emita o seguinte comando:

```
GRTMQMAUT OBJ('ObjectProfile
') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALL) MQMNAME('
QMgrName')
```

- Para z/OS, emita os comandos a seguir:

```
RDEFINE MQNLIST
QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile
 CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
```

Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas. No z/OS, este valor também pode ser o nome de um grupo de filas compartilhadas.

ObjectProfile
O nome do perfil de objeto ou do perfil genérico do qual alterar as autorizações.

GroupName
O nome do grupo que receberá acesso.

Autoridade para administrar IBM WebSphere MQ em sistemas UNIX, Linux, and Windows

Os administradores do IBM WebSphere MQ podem usar todos os comandos do IBM WebSphere MQ e conceder autoridades para outros usuários. Quando administradores emitem comandos para gerenciadores de filas remotas, ele devem ter a autoridade necessária no gerenciador de filas remotas. Considerações adicionais aplicam-se a sistemas Windows

IBM WebSphere MQ administradores têm autoridade para usar todos os comandos do WebSphere MQ (incluindo os comandos para conceder autoridades do WebSphere MQ para outros usuários)

Para ser um administrador do IBM WebSphere MQ , deve-se ser um membro de um grupo especial chamado grupo *mqm* (ou um membro do grupo Administradores em sistemas Windows). O grupo mqm é criado automaticamente quando o WebSphere MQ está instalado; inclua usuários adicionais no grupo para permitir que eles executem a administração Todos os membros desse grupo possuem acesso a todos os recursos. Esse acesso pode ser revogado somente com a remoção de um usuário do grupo mqm e emitindo o comando REFRESH SECURITY. Os administradores podem usar comandos de controle para administrar o WebSphere MQ. Um desses comandos de controle é **setmqaut**, que é usado para conceder autoridades a outros usuários para permitir que eles acessem ou controlem recursos do WebSphere MQ . Os comandos PCF para gerenciar os registros de autoridade ficam disponíveis a não administradores aos quais tenham sido concedidas autoridades dsp e chg no gerenciador de filas. Para obter mais informações sobre como gerenciar autoridades usando comandos PCF, consulte Formatos de comando programáveis.

Os administradores podem usar o comando de controle **runmqsc** para emitir comandos IBM WebSphere MQ Script (MQSC). Quando **runmqsc** é utilizado no modo indireto para enviar comandos MQSC para um gerenciador de fila remoto, cada comando será encapsulado dentro de um comando PCF Escape. Os administradores devem ter as autoridades requeridas para os comandos MQSC serem processados pelo gerenciador de fila remoto. O WebSphere MQ Explorer emite comandos PCF para executar tarefas de administração. Os administradores não requerem nenhuma autoridade adicional para usar o WebSphere MQ Explorer para administrar um gerenciador de filas no sistema local. Quando o IBM WebSphere MQ Explorer é usado para administrar um gerenciador de filas em outro sistema, os administradores devem ter as autoridades necessárias para que os comandos de PCF sejam processados pelo gerenciador de filas remoto.

Para obter mais informações sobre verificações de autoridade quando os comandos PCF e MQSC são processados, consulte os seguintes tópicos:

- Para comandos PCF que operam em gerenciadores de fila, filas, processos, listas de nomes e objetos de informações sobre autenticação, consulte Autoridade para trabalhar com objetos WebSphere MQ.

Consulte esta seção para obter os comandos MQSC equivalentes encapsulados nos comandos PCF Escape.

- Para comandos PCF que operam em canais, iniciadores de canais, listeners e clusters, consulte Segurança de Canal.
- Para comandos PCF que operam em registros de autoridade, consulte verificação de autoridade para comandos PCF

Além disso, em sistemas Windows , a conta SYSTEM tem acesso total aos recursos do WebSphere MQ

Em plataformas UNIX and Linux, um ID do usuário especial de mqm também é criado, para uso pelo produto apenas. Ele nunca deverá estar disponível para usuários não privilegiados. Todos os objetos do WebSphere MQ pertencem ao ID do usuário mqm.

Em sistemas Windows , os membros do grupo Administradores também podem administrar qualquer gerenciador de fila, como a conta SYSTEM. Também é possível criar um grupo mqm de domínio no controlador de domínio que contém todos os IDs de usuário privilegiado dentro do domínio e incluí-lo no grupo mqm local. Alguns comandos, por exemplo **crtmqm**, manipulam as autoridades nos objetos IBM WebSphere MQ e, portanto, precisam de autoridade para trabalhar com esses objetos (conforme descrito nas seções a seguir). Membros do grupo mqm têm autoridade para trabalhar com todos os objetos, mas pode haver circunstâncias nos sistemas Windows quando a autoridade é negada se você tiver um usuário local e um usuário autenticado por domínio com o mesmo nome. Isso é descrito no “Principais e Grupos” na página 204.

As versões do Windows com um recurso de Controle de Conta do Usuário (UAC) restringem as ações que os usuários podem executar em determinados recursos do sistema operacional, mesmo se forem membros do grupo Administradores. Se seu ID do usuário estiver no grupo Administradores, mas não no grupo mqm, você deverá usar um prompt de comandos elevado para emitir comandos do administrador do WebSphere MQ , como **crtmqm**, caso contrário, o erro "AMQ7077: Você não está autorizado a executar a operação solicitada" será gerado. Para abrir um prompt de comandos elevado, clique com o botão direito no item de menu inicial, ou ícone, para o prompt de comandos e selecione "Executar como administrador".

Não é necessário ser membro do grupo mqm para fazer o seguinte:

- Emita os comandos a partir de um programa de aplicativo que emite os comandos PCF ou comandos MQSC em um comando Escape PCF, a menos que os comandos manipulem os iniciadores de canal. (Esses comandos são descritos em “Protegendo Definições do Inicializador de Canais” na página 72).
- Emita as chamadas MQI a partir de um programa de aplicativo (a menos que você queira usar as ligações de caminho rápido na chamada `MQCONNX`).
- Use o comando `crtmqcvx` para criar um fragmento de código que executa conversão de dados em estruturas de tipo de dados.
- Usar o comando `dspmq` para exibir gerenciadores de filas.
- Use o comando `dspmqtrc` para exibir a saída de rastreio formatada do WebSphere MQ

Uma limitação de 12 caracteres aplica-se ao grupo e aos IDs do usuário.

As plataformas UNIX and Linux geralmente restringem o comprimento de um ID do usuário a 12 caracteres. AIX Versão 5.3 aumentou esse limite, mas o WebSphere MQ continua a observar uma restrição de 12 caracteres em todas as plataformas do UNIX and Linux . Se você usar um ID do usuário maior que 12 caracteres, o WebSphere MQ o substituirá pelo valor `UNKNOWN`. Não defina um ID do usuário com um valor de `UNKNOWN`.

Gerenciando o Grupo mqm

Os usuários no grupo mqm recebem privilégios administrativos completos sobre o WebSphere MQ. Por esta razão, não é necessário inscrever usuários comuns e de aplicativos no grupo mqm. O grupo mqm deve conter apenas as contas dos administradores do WebSphere MQ

Estas tarefas estão descritas em:

- Criando e Gerenciando Grupos no Windows

- Criando e Gerenciando Grupos no HP-UX
- Criando e Gerenciando Grupos no AIX
- Criando e Gerenciando Grupos no Solaris
- Criando e gerenciando grupos no Linux

Se seu controlador de domínio for executado no Windows 2000 ou Windows 2003, seu administrador de domínio poderá ter que configurar uma conta especial para o WebSphere MQ usar. Isso é descrito em Configurando as contas do WebSphere MQ

Autoridade para trabalhar com objetos IBM WebSphere MQ em sistemas UNIX, Linux, and Windows

Todos os objetos são protegidos pelo IBM WebSphere MQ e deve ser concedida autoridade apropriada aos diretores para acessá-los. Diferentes principais precisam de diferentes direitos de acesso a diferentes objetos.

Gerenciadores de filas, filas, definições de processos, listas de nomes, canais, canais de conexão do cliente, listeners, serviços e objetos de informações sobre autenticação são todos acessados a partir de aplicativos que usam chamadas MQI ou comandos PCF. Todos esses recursos são protegidos pelo WebSphere MQe os aplicativos precisam ter permissão para acessá-los.. A entidade que faz a solicitação pode ser um usuário, um programa de aplicativo que emite uma chamada MQI ou um programa de administração que emite um comando PCF. O identificador do solicitante é referido como o *principal*.

Diferentes grupos de principais podem receber diferentes tipos de autoridade de acesso ao mesmo objeto. Por exemplo, para uma fila específica, um grupo pode ter permissão para executar operações put e get; outro grupo pode ter permissão apenas para navegar na fila (MQGET com a opção de procura) De forma semelhante, alguns grupos podem ter as autoridades put e get para uma fila, mas podem não ter permissão para alterar atributos da fila ou para excluí-la.

Algumas operações são especialmente sigilosas e devem ser limitadas a usuários privilegiados. Por exemplo:

- Acesso a algumas filas especiais, como filas de transmissão ou a fila de comandos SYSTEM.ADMIN.COMMAND.QUEUE
- Execução de programas que usam opções de contexto completas de MQI
- Criação e exclusão de filas de aplicativos

A permissão de acesso total a um objeto é fornecida automaticamente ao ID do usuário que criou o objeto e a todos os membros do grupo mqm (e aos membros do grupo de Administradores locais nos sistemas Windows).

Conceitos relacionados

“Autoridade para administrar IBM WebSphere MQ em sistemas UNIX, Linux, and Windows” na página 200
Os administradores do IBM WebSphere MQ podem usar todos os comandos do IBM WebSphere MQ e conceder autoridades para outros usuários. Quando administradores emitem comandos para gerenciadores de filas remotas, ele devem ter a autoridade necessária no gerenciador de filas remotas. Considerações adicionais aplicam-se a sistemas Windows

Quando verificações de segurança são feitas em sistemas UNIX, Linux, and Windows

As verificações de segurança são feitas geralmente ao conectar-se a um gerenciador de filas, abrir ou fechar objetos e colocar ou obter mensagens.

As verificações de segurança feitas para um aplicativo típico são as seguintes:

Conectando-se ao gerenciador de filas (chamadas MQCONN ou MQCONNX)
Eta é a primeira vez que o aplicativo é associado a um determinado gerenciador de filas. O gerenciador de filas interroga o ambiente operacional para descobrir o ID do usuário associado ao

aplicativo. O WebSphere MQ então verifica se o ID do usuário está autorizado a se conectar ao gerenciador de filas e retém o ID do usuário para verificações futuras.

Os usuários não precisam se conectar ao WebSphere MQ; WebSphere MQ assume que os usuários se conectaram ao sistema operacional subjacente e foram autenticados por isso

Abrindo o objeto (chamadas MQOPEN ou MQPUT1)
WebSphere MQ objetos são acessados abrindo o objeto e emitindo comandos com relação a ele. Todas as verificações de recursos são executadas quando o objeto é aberto, em vez de quando ele é realmente acessado. Isso significa que a solicitação de **MQOPEN** deve especificar o tipo de acesso necessário (por exemplo, se o usuário deseja apenas pesquisar o objeto ou executar uma atualização, como colocar mensagens em uma fila).

WebSphere MQ verifica o recurso nomeado na solicitação **MQOPEN** . Para um objeto de fila de alias ou fila remota, a autorização usada é aquela do próprio objeto, não da fila à qual a fila do alias ou a fila remota é resolvida. Isso significa que o usuário não precisa de permissão para acessá-la. Limite a autoridade para criar filas a usuários privilegiados. Se você não fizer isto, os usuários podem efetuar bypass do controle de acesso normal simplesmente criando um alias. Se uma fila remota for referida explicitamente com ambos os nomes, da fila e do gerenciador de filas, a fila de transmissão associada ao gerenciador de filas remotas será verificada.

A autoridade para uma fila dinâmica baseia-se naquela da fila modelo da qual se deriva, mas não é necessariamente a mesma. Isso é descrito na Nota “1” na página 93.

O ID do usuário usado pelo gerenciador de filas para verificações de acesso é o ID do usuário obtido do ambiente operacional do aplicativo conectado ao gerenciador de filas. Um aplicativo adequadamente autorizado pode emitir uma chamada **MQOPEN** especificando um ID do usuário alternativo; verificações de controle de acesso são, então, feitas no ID do usuário alternativo. Isso não altera o ID do usuário associado ao aplicativo, apenas aquele usado para verificações de controle de acesso.

Colocando e obtendo mensagens (chamadas MQPUT ou MQGET)
Não são executadas verificações de controle de acesso.

Fechando o objeto (MQCLOSE)
Não são executadas verificações de controle de acesso, a menos que o **MQCLOSE** resulte na exclusão de uma fila dinâmica. Neste caso, há uma verificação para ver se o ID do usuário tem autorização para excluir a fila.

Inscrevendo-se em um tópico (MQSUB)
Quando um aplicativo é subscrito para um tópico, ele especifica o tipo de operação que precisa executar. Ele estará criando uma nova assinatura, alterando uma assinatura existente ou continuando uma assinatura existente sem alterá-la. Para cada tipo de operação, o gerenciador de filas verifica se o ID do usuário que está associado ao aplicativo possui a autoridade para executar a operação.

Quando um aplicativo assina um tópico, as verificações de autoridade são executadas com relação aos objetos do tópico que estão localizados na árvore de tópicos em, ou acima, o ponto na árvore de tópicos em que o aplicativo se inscreveu As verificações de autoridade podem envolver verificações em mais de um objeto de tópico.

O ID do usuário que o gerenciador de fila utiliza para as verificações de autoridade é o ID obtido do sistema operacional quando o aplicativo estabelece conexão com o gerenciador de fila.

O gerenciador de filas executa verificações de autoridade em filas de assinantes, mas não em filas gerenciadas.

Como o controle de acesso é implementado pelo IBM WebSphere MQ em sistemas UNIX, Linux, and Windows

O IBM WebSphere MQ usa os serviços de segurança fornecidos pelo sistema operacional subjacente, usando o gerenciador de autoridade de objeto. O IBM WebSphere MQ fornece comandos para criar e manter listas de controle de acesso.

Uma interface de controle de acesso chamada Authorization Service Interface faz parte do WebSphere MQ. O WebSphere MQ fornece uma implementação de um gerenciador de controle de acesso (em conformidade com a Interface de Serviço de Autorização) conhecido como o *object authority manager (OAM)*. Isso é automaticamente instalado e ativado para cada gerenciador de filas que você criar, a menos que você especifique de outra maneira, (conforme descrito em "Evitando verificações de acesso de segurança nos sistemas UNIX, Linux, and Windows" na página 170). O OAM pode ser substituído por qualquer usuário ou componente gravado do fornecedor que esteja em conformidade com a Interface de serviço de autorização.

O OAM explora os recursos de segurança do sistema operacional subjacente, usando IDs de usuário e de grupo do sistema operacional. Os usuários poderão acessar os objetos do WebSphere MQ somente se tiverem a autoridade correta "Controlando o acesso a objetos usando o OAM nos sistemas UNIX, Linux e Windows" na página 162 descreve como conceder e revogar essa autoridade.

O OAM mantém uma lista de controle de acesso (ACL) para cada recurso que controla. Os dados de autorização são armazenados em uma fila local chamada SYSTEM.AUTH.DATA.QUEUE. O acesso a essa fila é restrito a usuários no grupo mqm e, adicionalmente, no Windows, a usuários no grupo Administradores e usuários com login efetuado com o ID SYSTEM. O acesso de usuário à fila não pode ser alterado.

O WebSphere MQ fornece comandos para criar e manter listas de controle de acesso. Para obter informações adicionais sobre esses comandos, consulte "Controlando o acesso a objetos usando o OAM nos sistemas UNIX, Linux e Windows" na página 162.

O WebSphere MQ transmite ao OAM uma solicitação contendo um principal, um nome de recurso e um tipo de acesso O OAM concede ou rejeita o acesso com base na ACL que mantém. WebSphere MQ segue a decisão do OAM; se o OAM não puder tomar uma decisão, o WebSphere MQ não permitirá acesso.

Identificando o ID do usuário em sistemas UNIX, Linux, and Windows

O gerenciador de autoridade de objeto identifica o diretor que está solicitando acesso a um recurso. O ID do usuário usado como o principal varia de acordo com o contexto.

O gerenciador de autoridade de objeto (OAM) deve ser capaz de identificar quem está solicitando acesso a um recurso específico. O IBM WebSphere MQ usa o termo *diretor* para fazer referência a este identificador. O principal é estabelecido quando o aplicativo se conecta ao gerenciador de filas pela primeira vez; ele é determinado pelo gerenciador de filas a partir do ID do usuário associado ao aplicativo de conexão. (Se o aplicativo emitir chamadas XA sem se conectar ao gerenciador de filas, o ID do usuário associado ao aplicativo que emite a chamada xa_open será usado para verificações de autoridade pelo gerenciador de filas.)

Em sistemas UNIX and Linux, as rotinas de autorização verificam o ID do usuário real (efetuado log in) ou o ID do usuário efetivo associado ao aplicativo. O ID do usuário verificado pode depender do tipo de ligação. Para obter detalhes, consulte Serviços Instaláveis.

O IBM WebSphere MQ propagará o ID do usuário recebido do sistema no cabeçalho da mensagem (estrutura do MQMD) de cada mensagem como a identificação do usuário. Esse identificador faz parte das informações de contexto da mensagem e é descrito em "Autoridade de contexto em sistemas UNIX, Linux e Windows" na página 206. Os aplicativos não podem alterar essas informações, a não ser que tenham sido autorizados a alterar as informações de contexto.

Principais e Grupos

Principais podem pertencer a grupos. É possível conceder acesso a um recurso específico para grupos em vez de indivíduos, para reduzir a quantia de administração necessária. Em sistemas UNIX and Linux , todas as Listas de Controle de Acesso (ACLs) são baseadas em grupos, mas em sistemas Windows , os ACLS são baseados em IDs de usuário e grupos..

Por exemplo, é possível definir um grupo consistindo em usuários que desejam executar um determinado aplicativo. Outros usuários podem receber acesso a todos os recursos que precisam, incluindo seus IDs de usuário no grupo apropriado. Esse processo é descrito em:

- Criando e Gerenciando Grupos no Windows

- Criando e Gerenciando Grupos no HP-UX
- Criando e Gerenciando Grupos no AIX
- Criando e Gerenciando Grupos no Solaris
- Criando e gerenciando grupos no Linux

Um principal pode pertencer a mais de um grupo (seu conjunto de grupos). Ele tem a agregação de todas as autoridades concedidas a cada grupo em seu conjunto de grupos. Essas autoridades são armazenadas em cache, portanto, quaisquer mudanças feitas na associação ao grupo do principal não são reconhecidas até que o gerenciador de filas seja reiniciado, a menos que você emita o comando MQSC REFRESH SECURITY (ou o PCF equivalente).

Sistemas UNIX and Linux

Todas ACLs são baseadas em grupos. Quando um usuário tem acesso concedido a um recurso específico, o grupo primário do ID do usuário é incluído na ACL. O ID do usuário individual não é incluído e autoridade é concedida a todos os membros desse grupo. Por causa disso, observe que é possível, acidentalmente, mudar a autoridade de um diretor mudando a autoridade de outro diretor no mesmo grupo. Todos os usuários são nominalmente designados ao grupo de usuários padrão *nobody* e, por padrão, nenhuma autorização é fornecida a esse grupo. É possível alterar a autorização no grupo *nobody* para conceder acesso aos recursos do WebSphere MQ aos usuários sem autorizações específicas. .

Não defina um ID do usuário com o valor "UNKNOWN". O valor de "UNKNOWN" é usando quando um ID do usuário é muito longo, portanto os IDs de usuário arbitrário usam as autoridades de acesso de UNKNOWN.

Os IDs de usuário podem conter até 12 caracteres e nomes de grupos com até 12 caracteres.

Sistemas Windows

As ACLs baseiam-se em IDs de usuário e grupos. As verificações são as mesmas dos sistemas UNIX , exceto que os IDs de usuário individuais também podem ser exibidos na ACL. É possível ter usuários diferentes em domínios diferentes com o mesmo ID do usuário. WebSphere MQ permite que IDs de usuário sejam qualificados por um nome de domínio para que esses usuários possam receber diferentes níveis de acesso.

O nome do grupo pode, opcionalmente, incluir um nome de domínio, especificado nos formatos a seguir:

```
GroupName@domain
domain\GroupName
```

Grupos globais são marcadas pelo OAM somente em dois casos:

1. A sub-rotina de segurança do gerenciador de filas inclui a configuração: `GroupModel=GlobalGroups`; consulte Segurança.
2. O gerenciador de filas está usando um grupo de acesso de segurança alternativo; consulte **crtmqm**

Os IDs de usuário podem conter até 20 caracteres, nomes de domínio até 15 caracteres e nomes de grupos até 64 caracteres.

O OAM verifica, primeiramente, o banco de dados de segurança local, em seguida, o banco de dados de domínio primário e, finalmente, o banco de dados de qualquer domínio confiável. O primeiro ID do usuário encontrado é usado pelo OAM para verificação. Cada um desses IDs de usuário pode ter associações diferentes ao grupo em um determinado computador.

Alguns comandos de controle (por exemplo, `crtmqm`) alteram autoridades em objetos WebSphere MQ usando o gerenciador de autoridade de objeto (OAM). O OAM procura nos banco de dados de segurança na ordem fornecida no parágrafo precedente para determinar os direitos de autoridade de um determinado ID do usuário. Como resultado, a autoridade determinada pelo OAM pode substituir o fato de que um ID do usuário é um membro do grupo mqm local. Por exemplo, se você emitir o comando `crtmqm` a partir de um ID do usuário autenticado por um controlador de domínio que tenha associação do grupo mqm local por meio de um grupo global, o comando falhará se o sistema tiver um usuário local do mesmo nome que não esteja no grupo mqm local.

Identificadores de segurança (SIDs) do Windows

WebSphere MQ no Windows usa o SID no qual ele está disponível. Se um SID do Windows não for fornecido com um pedido de autorização, o WebSphere MQ identificará o usuário com base apenas no nome do usuário, mas isso poderá resultar na concessão de autoridade errada

Em sistemas Windows , o identificador de segurança (SID) é usado para complementar o ID do usuário.. O SID contém informações que identificam os detalhes completos da conta do usuário no banco de dados do Security account manager (SAM) do Windows no qual o usuário está definido. Quando uma mensagem é criada no WebSphere MQ para Windows, WebSphere MQ armazena o SID no descritor de mensagem. Quando o WebSphere MQ no Windows executa verificações de autorização, ele usa o SID para consultar as informações completas do banco de dados do SAM. (O banco de dados do SAM em que o usuário está definido deve estar acessível para que essa consulta seja bem-sucedida.)

Por padrão, se um SID do Windows não for fornecido com um pedido de autorização, o WebSphere MQ identificará o usuário com base no nome do usuário sozinho.. Ele faz isso procurando nos bancos de dados de segurança na seguinte ordem:

1. O banco de dados de segurança local
2. O banco de dados de segurança do domínio primário
3. O banco de dados de segurança de domínios confiáveis

Se o nome do usuário não for exclusivo, a autoridade incorreta do WebSphere MQ poderá ser concedida Para evitar esse problema, inclua um SID em cada solicitação de autorização; o SID é usado por WebSphere MQ para estabelecer credenciais do usuário.

Para especificar que todas as solicitações de autorização devem incluir um SID, use `regedit`. Configure SecurityPolicy como NTSIDsRequired.

Autoridade de usuário alternativo nos sistemas UNIX, Linux e Windows

É possível especificar que um ID do usuário possa usar a autoridade de outro usuário ao acessar um objeto WebSphere MQ . Isso é chamado de *autoridade de usuário alternativo*e é possível usá-lo em qualquer objeto do WebSphere MQ

A autoridade de usuário alternativo é essencial onde um servidor recebe solicitações de um programa e deseja assegurar que o programa possui a autoridade necessária para a solicitação. O servidor pode ter a autoridade necessária, mas precisa saber se o programa tem a autoridade para as ações solicitadas.

Por exemplo, suponha que um programa do servidor em execução com um ID do usuário PAYSERV recupere uma mensagem de solicitação de uma fila que foi colocada na fila pelo ID do usuário USER1. Quando o programa do servidor recebe a mensagem de solicitação, ele processa a solicitação e coloca a resposta de volta na fila de resposta especificada com a mensagem de solicitação. Em vez de usar seu próprio ID do usuário (PAYSERV) para autorizar a abertura da fila de resposta, o servidor pode especificar um ID do usuário diferente, neste caso, USER1. Neste exemplo, é possível usar a autoridade de usuário alternativo para controlar se PAYSERV tem permissão para especificar USER1 como um ID do usuário alternativo ao abrir a fila de resposta.

O ID do usuário alternativo é especificado no campo **`AlternateUserId`** do descritor de objeto.

Autoridade de contexto em sistemas UNIX, Linux e Windows

Contexto são informações que se aplicam a uma determinada mensagem e está contido no descritor de mensagens, MQMD, que faz parte da mensagem. Aplicativos podem especificar os dados de contexto quando uma chamada `MQOPEN` ou `MQPUT` é feita.

As informações de contexto são fornecidas em duas seções:

Seção de Identidade
De quem a mensagem veio. Ela consiste nos campos `UserIdentifier`, `AccountingToken` e `ApplIdentityData`.

Seção de Origem
De onde a mensagem veio, e quando ela foi colocada na fila. Ela consiste nos campos PutApplType, PutApplName, PutDate, PutTime e ApplOriginData.

Aplicativos podem especificar os dados de contexto quando uma chamada MQOPEN ou MQPUT é feita. Esses dados podem ser gerados pelo aplicativo, passados de outra mensagem ou gerados pelo gerenciador de filas, por padrão. Por exemplo, dados de contexto podem ser usados por programas do servidor para verificar a identidade do solicitante, testar se a mensagem veio de um aplicativo em execução com um ID do usuário autorizado.

Um programa do servidor pode usar UserIdentifier para determinar o ID do usuário de um usuário alternativo. Use a autorização de contexto para controlar se o usuário pode especificar qualquer uma das opções de contexto em qualquer chamada MQOPEN ou MQPUT1.

Consulte Controlando informações de contexto para obter informações sobre as opções de contexto, e Visão geral para MQMD para descrições dos campos do descritor de mensagens relacionadas ao contexto.

Implementando o controle de acesso em saídas de segurança

É possível implementar o controle de acesso em uma saída de segurança usando o MCAUserIdentifier ou o gerenciador de autoridade de objeto.

MCAUserIdentifier

Cada instância de um canal atual tem uma estrutura de definição de canais associada, MQCD. Os valores iniciais dos campos em MQCD são determinados pela definição de canal criada por um administrador do WebSphere MQ . Em particular, o valor inicial de um dos campos, *MCAUserIdentifier* , é determinado pelo valor do parâmetro MCAUSER no comando DEFINE CHANNEL ou pelo equivalente a MCAUSER, se a definição de canais for criada de outra forma. *MCAUserIdentifier* contém os primeiros 12 bytes do identificador de usuário MCA. Se o identificador de usuários do MCA não estiver em branco, ele especificará o identificador de usuários a ser usado pelo agente do canal de mensagem para autorização para acessar recursos do MQ . Assegure-se de que o MCAUSER tenha menos de 12 caracteres na plataforma Windows.

A estrutura MQCD passa para um programa de saída de canal quando é chamada por um MCA. Quando chamada, a saída de segurança pode alterar o valor de *MCAUserIdentifier*, substituindo qualquer valor especificado na definição de canais.

Nos sistemas IBM i, UNIX, Linux e Windows , a menos que o valor de *MCAUserIdentifier* esteja em branco, o gerenciador de filas usa o valor de *MCAUserIdentifier* como o ID do usuário para verificações de autoridade quando um MCA tenta acessar os recursos do gerenciador de filas após ter se conectado ao gerenciador de filas. Se o valor de *MCAUserIdentifier* estiver em branco, o gerenciador de fila utilizará então o ID do usuário padrão do MCA. Isso se aplica aos canais RCVR, RQSTR, CLUSRCVR e SVRCONN. Para MCAs de envio, o ID do usuário padrão sempre é usado para verificações de autoridade, mesmo que o valor de *MCAUserIdentifier* não esteja em branco.

No z/OS, o gerenciador de filas pode usar o valor de *MCAUserIdentifier* para verificações da autoridade, desde que não esteja em branco. Para MCAs de recepção e MCAs de conexão do servidor, o gerenciador de fila utiliza o valor *MCAUserIdentifier* para verificações de autoridade dependendo:

- Do valor do parâmetro PUTAUT na definição do canal
- O perfil RACF usado para as verificações
- Do nível de acesso do ID do usuário do espaço de endereço do inicializador de canais do perfil RESLEVEL

Para MCAs de envio, depende:

- De o MCA ser um originador da chamada ou um receptor
- Do nível de acesso do ID do usuário do espaço de endereço do inicializador de canais do perfil RESLEVEL

O ID do usuário que uma saída de usuário armazena no *MCAUserIdentifier* pode ser adquirido de várias formas. Estes são alguns exemplos:

- Desde que não haja saída de segurança na extremidade do cliente de um canal MQI, um ID do usuário associado ao aplicativo cliente do WebSphere MQ flui da conexão do cliente MCA para a conexão do servidor MCA quando o aplicativo cliente emite uma chamada MQCONN.O MCA da conexão do servidor armazena este ID do usuário no campo *RemoteUserIdentifier* na estrutura de definição do canal, MQCD. Se o valor de *MCAUserIdentifier* estiver em branco nesse momento, o MCA armazenará o mesmo ID do usuário no *MCAUserIdentifier*. Se o MCA não armazenar o ID do usuário em *MCAUserIdentifier*, uma saída de segurança poderá fazê-lo posteriormente configurando *MCAUserIdentifier* com o valor de *RemoteUserIdentifier*.

  Se o ID do usuário que flui do sistema do cliente estiver entrando em um novo domínio de segurança e não for válido no sistema do servidor, a saída de segurança poderá substituir o ID do usuário por um que seja válido e armazenar o ID do usuário substituto no *MCAUserIdentifier*.
- O ID do usuário pode ser enviado pela saída de segurança do parceiro em uma mensagem de segurança.

  Em um canal de mensagens, uma saída de segurança chamada pelo MCA de envio pode enviar o ID do usuário sob o qual o MCA de envio está sendo executado. Uma saída de segurança chamada pelo MCA receptor pode então armazenar o ID do usuário no *MCAUserIdentifier* . Da mesma forma, em um canal MQI, uma saída de segurança na extremidade do cliente do canal pode enviar o ID do usuário associado ao aplicativo cliente MQI WebSphere MQ . Uma saída de segurança na parte do servidor do canal pode então armazenar o ID do usuário no *MCAUserIdentifier*. Como no exemplo anterior, se o ID do usuário não for válido no sistema de destino, a saída de segurança pode substituir o ID do usuário por um que seja válido e armazenar o substituto no *MCAUserIdentifier*.

  Se um certificado digital foi recebido como parte do serviço de identificação e autenticação, uma saída de segurança poderá mapear o Nome Distinto no certificado para um ID do usuário que seja válido no sistema de destino. Ele pode então armazenar o ID do usuário no *MCAUserIdentifier*.
- Se for usado SSL no canal, o DN (Distinguished Name) do parceiro será passado para a saída no campo SSLPeerNamePtr do MQCD, e o DN do emissor do certificado é passado para a saída no campo SSLRemCertIssNamePtr do MQCXP.

Para obter mais informações sobre o campo *MCAUserIdentifier*, a estrutura de definição de canal, MQCD e a estrutura de parâmetros de saída de canal, MQCXP, consulte Chamadas de saída do canal e estrutura de dados. Para obter mais informações sobre o ID do usuário que flui de um sistema do cliente em um canal MQI, consulte Controle de Acesso.

Nota: Os aplicativos de saída de segurança construídos antes da liberação do WebSphere MQ v7.1 podem requerer atualização Para obter mais informações, consulte Programas de saída de segurança de canal.

WebSphere MQ autenticação do usuário do gerenciador de autoridade de objeto

No WebSphere MQ conexões do cliente MQI, saídas de segurança podem ser usadas para modificar ou criar a estrutura MQCSP usada na autenticação do usuário do gerenciador de autoridade de objeto (OAM). Isso é descrito em Programas de saída do canal para canais do sistema de mensagens

Implementando controle de acesso em saídas de mensagem

Talvez seja necessário usar uma saída de mensagem para substituir um ID do usuário por outro.

Considere um aplicativo cliente que envia uma mensagem para um aplicativo do servidor. O aplicativo do servidor pode extrair o ID do usuário do campo *UserIdentifier* no descritor de mensagens e, desde que ele tenha autoridade de usuário alternativa, solicitar que o gerenciador de filas use esse ID do usuário para verificações de autoridade quando ele acessar recursos do WebSphere MQ em nome do cliente.

Se o parâmetro PUTAUT for configurado como CTX (ou ALTMCA no z/OS) na definição do canal, o ID do usuário no campo *UserIdentifier* de cada mensagem recebida será usado para verificações de autoridade quando o MCA abrir a fila de destino.

Em certas circunstâncias, quando uma mensagem de relatório é gerada, é colocada utilizando a autoridade do ID do usuário no campo *UserIdentifier* da mensagem que está causando o relatório. Em particular, os relatórios COD (confirm-on-delivery) e relatórios de expiração são sempre colocados com essa autoridade.

Em decorrência dessas situações, será necessário substituir um ID do usuário por outro no campo *UserIdentifier* à medida que uma mensagem entrar em um novo domínio de segurança. Isso pode ser feito por uma saída de mensagem na extremidade receptora do canal. Alternativamente, você pode garantir que o ID do usuário no campo *UserIdentifier* de uma mensagem de entrada seja definido no novo domínio de segurança.

Se uma mensagem de entrada contiver um certificado digital para o usuário do aplicativo que enviou a mensagem, uma saída de mensagem poderá validar o certificado e mapear o Nome Distinto no certificado para um ID do usuário que seja válido no sistema receptor. Ela poderá definir o campo *UserIdentifier* no descritor de mensagem desse ID do usuário.

Se for necessário que uma saída de mensagem altere o valor do campo *UserIdentifier* em uma mensagem de entrada, poderá ser apropriado que a saída autentique o emissor da mensagem ao mesmo tempo. Para obter mais detalhes, consulte "Mapeamento de identidade em saídas de mensagem" na página 151.

Implementando o controle de acesso na saída de API e saída cruzada da API

Uma saída de cruzamento de API ou API pode fornecer controles de acesso para complementar aqueles fornecidos pelo WebSphere MQ. Especificamente, a saída pode fornecer controle de acesso no nível de mensagem. A saída pode assegurar que um aplicativo coloque ou obtenha de uma fila somente mensagens que satisfaçam determinados critérios.

Considere os seguintes exemplos:

- Uma mensagem contém informações sobre um pedido. Quando um aplicativo tenta colocar uma mensagem em uma fila, uma saída API ou saída cruzada da API pode verificar se o valor total do pedido é menor do que algum limite prescrito.
- Mensagens chegam a uma fila de destino a partir de gerenciadores de filas remotos. Quando um aplicativo tenta obter uma mensagem da fila, uma saída API ou saída cruzada da API pode verificar se o emissor da mensagem está autorizado a enviar uma mensagem para a fila.

Confidencialidade das mensagens

Para manter a confidencialidade, criptografe suas mensagens. Há vários métodos de criptografia de mensagens no WebSphere MQ , dependendo de suas necessidades

Sua opção de CipherSpec determina o nível de confidencialidade que você possui.

Se você precisar de proteção de dados de ponta a ponta no nível do aplicativo para sua infraestrutura do sistema de mensagens ponto a ponto, poderá usar o WebSphere MQ Advanced Message Security para criptografar as mensagens ou gravar sua própria saída de API ou saída cruzada da API.

Se você precisar criptografar mensagens somente enquanto elas estão sendo transportadas através de um canal, porque você tem a segurança adequada em seus gerenciadores de filas, é possível usar o SSL ou TLS ou é possível escrever sua própria saída de segurança, saída de mensagem ou enviar e receber programas de saída.

Para obter mais informações sobre o WebSphere MQ Advanced Message Security, consulte "Planejamento do Advanced Message Security" na página 65.O uso de SSL e TLS com o WebSphere MQ é descrito em "IBM WebSphere MQ Suporte para SSL e TLS" na página 23 O uso de programas de saída em criptografia de mensagem é descrito em "Implementando confidencialidade em programas de saída do usuário" na página 229.

Conectando dois gerenciadores de filas usando SSL ou TLS

As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

Para configurar sua instalação de SSL ou TLS, você deve definir seus canais para usar SSL ou TLS. Você também deve obter e gerenciar seus certificados digitais. Em um sistema de teste, é possível usar certificados autoassinados ou certificados emitidos por uma autoridade de certificação (CA) local. Em um sistema de produção, não use certificados autoassinados. Para obter mais informações, consulte ..// zs14140_.dita.

Para obter informações completas sobre como criar e gerenciar certificados, consulte “Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows” na página 116.

Esta coleção de tópicos apresenta as tarefas envolvidas na configuração de comunicações SSL e fornece orientação passo a passo sobre como concluir as tarefas.

Você também pode querer testar a autenticação de cliente SSL ou TLS, que são uma parte opcional dos protocolos. Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação do WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente.

Notes:

1. Neste contexto, um cliente SSL refere-se à conexão que está iniciando o handshake.
2. Consulte o Glossário para obter detalhes adicionais

Nos sistemas UNIX, Linux e Windows , o cliente SSL ou TLS envia um certificado somente se ele tiver um rotulado no formato correto do WebSphere MQ , que é `ibmwebspheremq` seguido pelo nome do seu gerenciador de filas alterado para minúsculas. Por exemplo, para `QM1`, `ibmwebspheremqqm1`.

WebSphere MQ usa o prefixo `ibmwebspheremq` em um rótulo para evitar confusão com certificados para outros produtos. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falha se a extremidade do canal que está agindo como o servidor SSL ou TLS estiver definida com o parâmetro SSLCAUTH configurado para REQUIRED ou um conjunto de valores do parâmetro SSLPEER. Para obter informações adicionais sobre como conectar um gerenciador de filas de forma anônima ou seja, quando o cliente SSL ou TLS não enviar um certificado, consulte “Conectando dois gerenciadores de filas usando autenticação unilateral” na página 214.

Usando certificados autoassinados para autenticação mútua de dois gerenciadores de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre dois gerenciadores de filas, usando certificados SSL ou TLS autoassinados.

Sobre esta tarefa

Cenário:

- Você possui dois gerenciadores de filas, QM1 e QM2, que precisam se comunicar com segurança. É necessário que a autenticação mútua seja executada entre QM1 e QM2.
- Você decidiu testar a comunicação segura usando certificados autoassinados.

A configuração resultante é semelhante à seguinte:

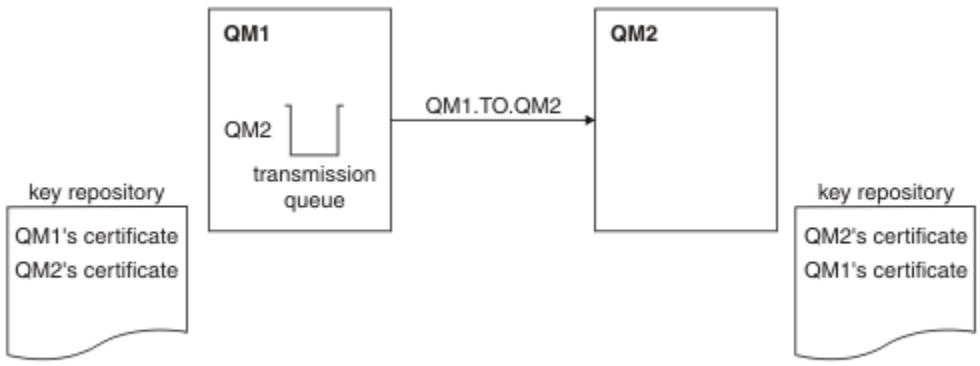


Figura 14. Configuração resultante desta tarefa

Na Figura 14 na página 211, o repositório de chaves para QM1 contém os certificados para QM1 e o certificado público de QM2. O repositório de chaves para QM2 contém o certificado para QM2 e o certificado público de QM1.

Procedimento

1. Prepare o repositório de chaves em cada gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Crie um certificado autoassinado para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
3. Extraia uma cópia de cada certificado:
   - Em UNIX, Linuxe Windows sistemas
4. Transfira a parte pública do certificado QM1 para o sistema QM2 e vice-versa, usando um utilitário como FTP.
5. Inclua o certificado do parceiro no repositório de chaves para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
6. No QM1, defina um canal emissor e a fila de transmissão associada, emitindo os comandos como o seguinte exemplo:

   ```
   DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
   SSLCIPH(RC4_MD5_US) DESCR('Sender channel using SSL from QM1 to QM2')

   DEFINE QLOCAL(QM2) USAGE(XMITQ)
   ```

   Este exemplo usa CipherSpec RC4_MD5. Os CipherSpecs em cada extremidade do canal devem ser iguais.
7. No QM2, defina um canal receptor, emitindo um comando como o seguinte exemplo:

   ```
   DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from QM1 to QM2')
   ```

   O canal deve ter o mesmo nome que o canal emissor definido na etapa 6 e usar o mesmo CipherSpec.
8. Inicie o canal.

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 14 na página 211

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM1.TO.QM2)                     CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=QM2,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

No gerenciador de filas QM2, insira o seguinte comando:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM2.TO.QM1)                     CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QM1)
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

Em cada caso, o valor de SSLPEER deve corresponder àquele do DN no certificado parceiro que foi criado na Etapa 2. O nome do emissor corresponde ao nome do peer porque o certificado é autoassinado.

SSLPEER é opcional. Se for especificado, o valor deve ser configurado para que o DN no certificado do parceiro (criado na etapa 2) seja permitido. Para obter mais informações sobre o uso de SSLPEER, consulte WebSphere MQ regras para valores SSLPEER .

Usando certificados assinados por CA para autenticação mútua de dois gerenciadores de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre dois gerenciadores de filas, usando certificados SSL ou TLS assinados por CA.

Sobre esta tarefa

Cenário:

- Você possui dois gerenciadores de filas, QMA e QMB, que precisam se comunicar com segurança. É necessário executar a autenticação mútua entre QMA e QMB.
- No futuro, você está planejando usar essa rede em um ambiente de produção e, portanto, decidiu usar os certificados assinados por CA desde o início.

A configuração resultante é semelhante à seguinte:

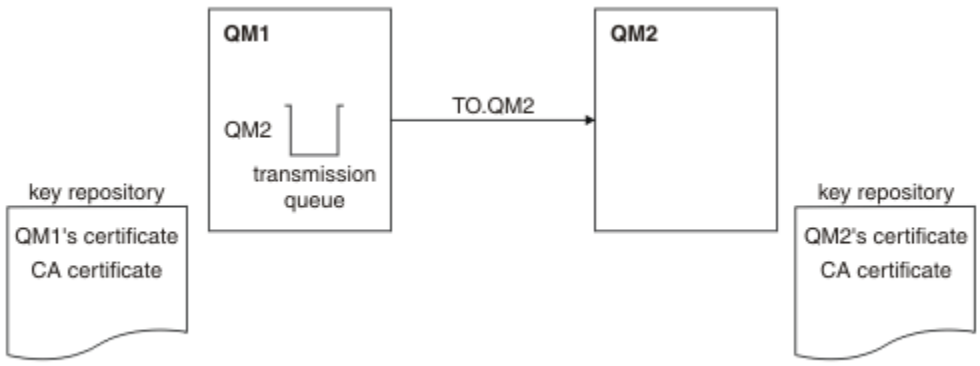


Figura 15. Configuração resultante desta tarefa

Na Figura 15 na página 213, o repositório de chaves para QMA contém o certificado de QMA e o certificado de CA. O repositório de chaves para QMB contém o certificado de QMB e o certificado de CA. Neste exemplo, o certificado de QMA e o certificado de QMB foram emitidos pela mesma CA. Se o certificado de QMA e o certificado de QMB foram emitidos por CAs diferentes, os repositórios de chaves para QMA e QMB deverão conter ambos os certificados de CA.

Procedimento

1. Prepare o repositório de chaves em cada gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Solicite um certificado assinado por CA para cada gerenciador de filas.

   É possível usar CAs diferentes para os dois gerenciadores de filas.
   - Em UNIX, Linuxe Windows sistemas
3. Inclua o certificado de Autoridade de certificação no repositório de chaves para cada gerenciador de filas:

   Se os Gerenciadores de filas estiverem usando Autoridades de certificação diferentes, o certificado de CA para cada Autoridade de certificação deverá ser incluído nos dois repositórios de chaves.
   - Em UNIX, Linuxe Windows sistemas
4. Incluir o certificado assinado por CA no repositório de chaves para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
5. No QMA, defina um canal emissor e a fila de transmissão associada emitindo comandos como o seguinte exemplo:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
   SSLCIPH(RC2_MD5_EXPORT) DESCR('Sender channel using SSL from QMA to QMB')

   DEFINE QLOCAL(QMB) USAGE(XMITQ)
   ```

   Este exemplo usa CipherSpec RC4_MD5. Os CipherSpecs em cada extremidade do canal devem ser iguais.
6. No QMB, defina um canal receptor, emitindo um comando como o exemplo a seguir:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
   ```

   O canal deve ter o mesmo nome que o canal emissor definido na etapa 6 e usar o mesmo CipherSpec.
7. Inicie o canal:

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 15 na página 213.

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir.

No gerenciador de filas QMA, insira o seguinte comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QMB)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QMB)
```

No gerenciador de filas QMB, insira o seguinte comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

Em cada caso, o valor de SSLPEER deve corresponder àquele do Nome Distinto (DN) no certificado parceiro que foi criado na Etapa 2. O nome do emissor corresponde ao DN do sujeito do certificado CA que assinou o certificado pessoal incluído na Etapa 4.

Conectando dois gerenciadores de filas usando autenticação unilateral

Siga estas instruções de amostra para modificar um sistema com autenticação mútua para permitir que um gerenciador de filas se conecte a outro usando autenticação unilateral; ou seja, quando o cliente SSL ou TLS não enviar um certificado.

Sobre esta tarefa

Cenário:

- Os dois gerenciadores de filas (QM1 e QM2) foram configurados conforme descrito em “Usando certificados assinados por CA para autenticação mútua de dois gerenciadores de filas” na página 212.
- Você deseja alterar o QM1 para que ele se conecte ao QM2 usando autenticação unilateral.

A configuração resultante é semelhante à seguinte:

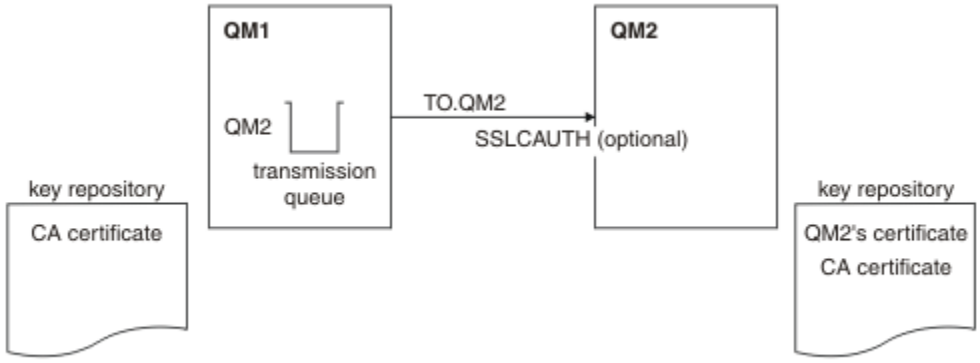


Figura 16. Gerenciadores de filas que permitem autenticação unilateral

Procedimento

1. Remova o certificado pessoal do QM1de seu repositório de chaves, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas O certificado é rotulado como a seguir:
     - `ibmwebspheremq` seguido pelo nome de seu gerenciador de filas dobrado para letras minúsculas Por exemplo, para `QM1` , `ibmwebspheremqqm1`..
2. Opcional: Em QM1, se qualquer canal SSL ou TLS tiver sido executado anteriormente, atualize o ambiente SSL ou TLS.
3. Permita conexões anônimas no

Resultados

Os repositórios de chaves e canais são mudados conforme ilustrado na Figura 16 na página 215

Como proceder a seguir

Se o canal emissor estava em execução e você emitiu o comando REFRESH SECURITY TYPE(SSL) (na etapa 2), o canal será reiniciado automaticamente. Se o canal emissor não estiver em execução, inicie-o.

Na extremidade do canal do servidor, a presença do valor de parâmetro do nome do peer na exibição de status do canal indica que um certificado de cliente foi emitido.

Verifique se a tarefa foi concluída com sucesso, emitindo alguns comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir:

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
```

```
   XMITQ(QM2)
```

No gerenciador de filas QM2, insira o seguinte comando:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)                            SSLCERTI( )
   SSLPEER( )                              STATUS(RUNNING)
   SUBSTATE(RECEIVE)                       XMITQ( )
```

No QM2, o campo SSLPEER está vazio, mostrando que o QM1 não enviou um certificado. No QM1, o valor de SSLPEER corresponde ao valor do DN no certificado pessoal do QM2.

Conectando um Cliente a um Gerenciador de Filas de Forma Segura

As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

Para configurar sua instalação de SSL ou TLS, você deve definir seus canais para usar SSL ou TLS. Você também deve obter e gerenciar seus certificados digitais. Em um sistema de teste, é possível usar certificados autoassinados ou certificados emitidos por uma autoridade de certificação (CA) local. Em um sistema de produção, não use certificados autoassinados. Para obter mais informações, consulte ..// zs14140_.dita.

Para obter informações completas sobre como criar e gerenciar certificados, consulte “Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows” na página 116.

Esta coleção de tópicos apresenta as tarefas envolvidas na configuração de comunicações SSL e fornece orientação passo a passo sobre como concluir as tarefas.

Você também pode querer testar a autenticação de cliente SSL ou TLS, que são uma parte opcional dos protocolos. Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação do WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente.

Nos sistemas UNIX, Linux, and Windows , o cliente SSL ou TLS envia um certificado somente se ele tiver um rotulado no formato correto do WebSphere MQ , que é `ibmwebspheremq` seguido por seu ID do usuário de logon alterado para minúsculas, por exemplo, `ibmwebspheremqmyuserid`.

WebSphere MQ usa o prefixo `ibmwebspheremq` em um rótulo para evitar confusão com certificados para outros produtos. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falha se a extremidade do canal que está agindo como o servidor SSL ou TLS estiver definida com o parâmetro SSLCAUTH configurado para REQUIRED ou um conjunto de valores do parâmetro SSLPEER. Para obter informações adicionais sobre como conectar a um gerenciador de filas de forma anônima, consulte “Conectando um cliente a um gerenciador de filas de forma anônima” na página 220.

Usando certificados autoassinados para autenticação mútua de um cliente e de um gerenciador de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre um cliente e um gerenciador de filas, usando certificados SSL ou TLS autoassinados.

Sobre esta tarefa

Cenário:

- Você tem um cliente, C1, e um gerenciador de filas, QM1, que precisam se comunicar de forma segura. É necessário executar a autenticação mútua entre C1 e QM1.
- Você decidiu testar a comunicação segura usando certificados autoassinados.

O DCM no IBM i não suporta certificados autoassinados, portanto, esta tarefa não é aplicável em sistemas IBM i.

A configuração resultante é semelhante à seguinte:

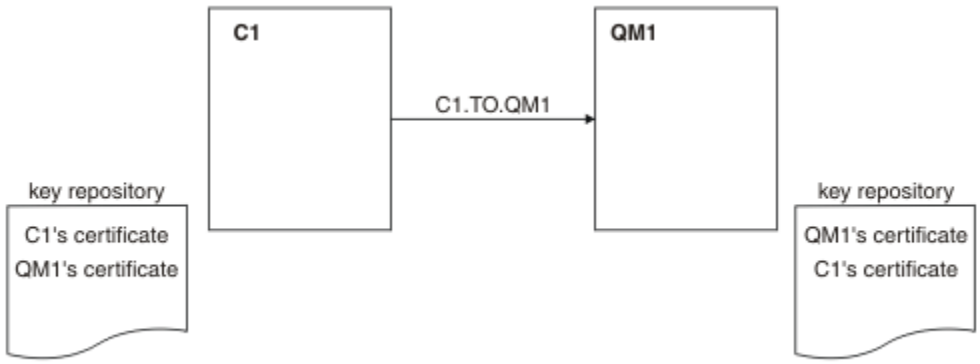


Figura 17. Configuração resultante desta tarefa

Na Figura 17 na página 217, o repositório de chaves para QM1 contém o certificado para QM1 e o certificado público de C1. O repositório de chaves para C1 contém o certificado para C1 e o certificado público de QM1.

Procedimento

1. Prepare o repositório de chaves no cliente e no gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Crie certificados autoassinados para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
3. Extraia uma cópia de cada certificado:
   - Em UNIX, Linuxe Windows sistemas
4. Transfira a parte pública do certificado C1 para o sistema QM1 e vice-versa, usando um utilitário como FTP.
5. Inclua o certificado de parceiro no repositório de chaves para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
6. Emita o comando REFRESH SECURITY TYPE(SSL) no gerenciador de filas.
7. Defina um canal de conexão do cliente de uma das seguintes maneiras:
   - Usando a chamada MQCONNX com a estrutura MQSCO no C1, conforme descrito em Criando um canal de conexão do cliente no cliente MQI do WebSphere MQ.
   - Usando uma tabela de definição de canal de cliente, conforme descrito em Criando definições de conexão do servidor e de conexão do cliente no servidor ..
8. No QM1, defina um canal de conexão do servidor, emitindo um comando como o seguinte exemplo:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
```

O canal deve ter o mesmo nome que o canal de conexão do cliente definido na etapa 6 e usar o mesmo CipherSpec.

Resultados

Repositórios de chaves e canais são criados conforme ilustrado em Figura 17 na página 217

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                       CURRENT
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                           SUBSTATE(RECEIVE)
```

É opcional configurar o atributo de filtro SSLPEER das definições de canal. Se a definição de canal SSLPEER estiver configurada, seu valor deverá corresponder ao DN do sujeito no certificado parceiro que foi criado na Etapa 2. Após uma conexão bem-sucedida, o campo SSLPEER na saída DISPLAY CHSTATUS mostra o DN do assunto do certificado de cliente remoto

Usando certificados assinados por CA para autenticação mútua de um cliente e de um gerenciador de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre um cliente e um gerenciador de filas, usando certificados SSL ou TLS assinados por CA.

Sobre esta tarefa

Cenário:

- Você tem um cliente, C1, e um gerenciador de filas, QM1, que precisam se comunicar de forma segura. É necessário executar a autenticação mútua entre C1 e QM1.
- No futuro, você está planejando usar essa rede em um ambiente de produção e, portanto, decidiu usar os certificados assinados por CA desde o início.

A configuração resultante é semelhante à seguinte:

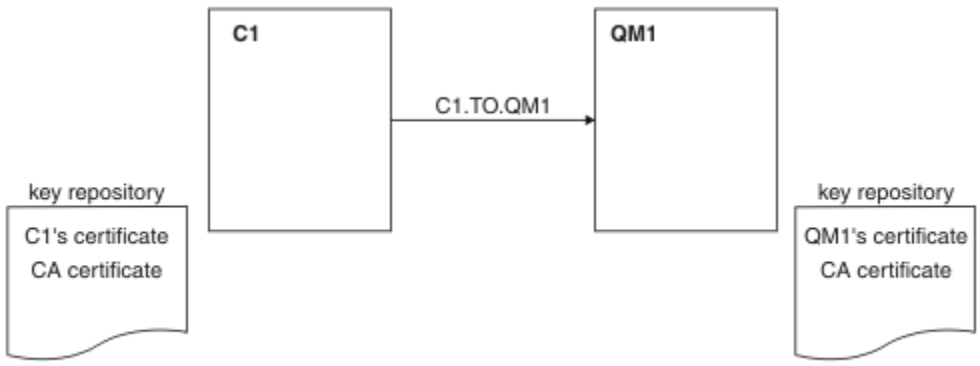


Figura 18. Configuração resultante desta tarefa

Na Figura 18 na página 219, o repositório de chaves para C1 contém certificado para C1 e o certificado de CA. O repositório de chaves para QM1 contém o certificado para QM1 e o certificado de CA. Neste exemplo, o certificado de C1 e o certificado de QM1 foram emitidos pela mesma CA. Se o certificado de C1 e o certificado de QM1 foram emitidos por CAs diferentes, os repositórios de chaves para C1 e QM1 deverão conter certificados de CA.

Procedimento

1. Prepare o repositório de chaves no cliente e no gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Solicite um certificado assinado por CA para o cliente e o gerenciador de filas.

   É possível usar CAs diferentes para o cliente e o gerenciador de filas.
   - Em UNIX, Linuxe Windows sistemas
3. Inclua o certificado de autoridade de certificação no repositório de chaves para o cliente e gerenciador de filas.

   Se o cliente e o gerenciador de filas estiverem usando diferentes Autoridades de certificação, o certificado de CA para cada Autoridade de certificação deverá ser incluído nos dois repositórios de chaves.
   - Em UNIX, Linuxe Windows sistemas
4. Inclua o certificado assinado por CA no repositório de chaves para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
5. Defina um canal de conexão do cliente de uma das seguintes maneiras:
   - Usando a chamada MQCONNX com a estrutura MQSCO no C1, conforme descrito em Criando um canal de conexão do cliente no cliente MQI do WebSphere MQ.
   - Usando uma tabela de definição de canal de cliente, conforme descrito em Criando definições de conexão do servidor e de conexão do cliente no servidor ..
6. No QM1, defina um canal de conexão do servidor emitindo um comando como o seguinte exemplo:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   O canal deve ter o mesmo nome que o canal de conexão do cliente definido na etapa 6 e usar o mesmo CipherSpec.

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 18 na página 219.

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

O campo SSLPEER na saída DISPLAY CHSTATUS mostra o DN do sujeito do certificado cliente remoto que foi criado na Etapa 2. O nome do emissor corresponde ao DN do sujeito do certificado CA que assinou o certificado pessoal incluído na Etapa 4.

Conectando um cliente a um gerenciador de filas de forma anônima

Siga estas instruções de amostra para modificar um sistema com autenticação mútua para permitir que um gerenciador de filas se conecte a outro de forma anônima.

Sobre esta tarefa

Cenário:

- O gerenciador de filas e o cliente (QM1 e C1) foram configurados conforme descrito em “Usando certificados assinados por CA para autenticação mútua de um cliente e de um gerenciador de filas” na página 218.
- Você deseja alterar C1 para que ele se conecte de forma anônima a QM1.

A configuração resultante é semelhante à seguinte:

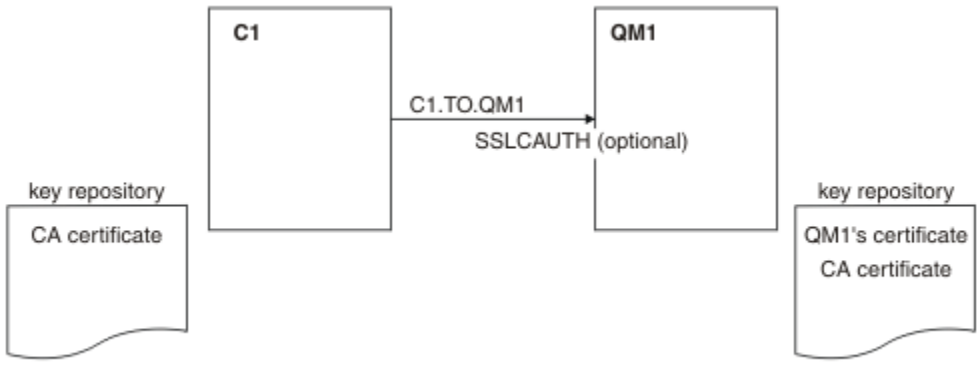


Figura 19. Cliente e gerenciador de filas permitindo conexão anônima

Procedimento

1. Remova o certificado pessoal do repositório de chaves para C1, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas O certificado é rotulado como a seguir:

– `ibmwebspheremq` seguido por seu ID do usuário de logon dobrado para minúsculas, por exemplo `ibmwebspheremqmyuserid`.

2. Reinicie o aplicativo cliente ou faça o aplicativo cliente fechar e reabrir todas as conexões SSL ou TLS.
3. Permita conexões anônimas no gerenciador de filas, emitindo o seguinte comando:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Resultados

Os repositórios de chaves e canais são mudados conforme ilustrado na Figura 19 na página 220

Como proceder a seguir

Na extremidade do canal do servidor, a presença do valor de parâmetro do nome do peer na exibição de status do canal indica que um certificado de cliente foi emitido.

Verifique se a tarefa foi concluída com sucesso, emitindo alguns comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir:

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI( )                             SSLPEER( )
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

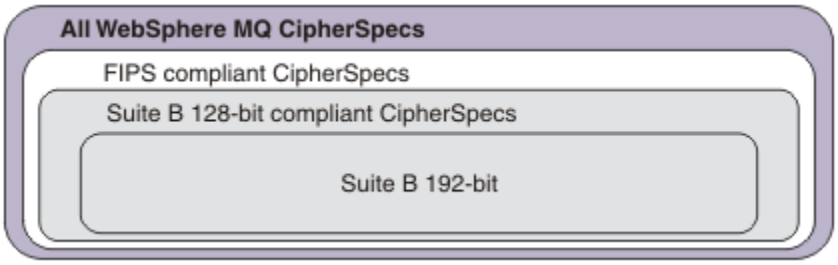
Os campos SSLCERTI e SSLPEER estão vazios, mostrando que o C1 não enviou um certificado.

Especificando CipherSpecs

Especifique um CipherSpec usando o parâmetro **SSLCIPH** no comando MQSC **DEFINE CHANNEL** ou no comando MQSC **ALTER CHANNEL** .

Alguns dos CipherSpecs que podem ser usados com IBM WebSphere MQ são compatíveis com FIPS. Outros, como `NULL_MD5`, não. Da mesma forma, alguns dos CipherSpecs compatíveis com FIPS também são compatíveis com o Conjunto B, embora outros não sejam. Todos os CipherSpecs compatíveis com o Conjunto B também são compatíveis com FIPS. Todos os CipherSpecs compatíveis com o Suite B caem em dois grupos: 128 bit (por exemplo, `ECDHE_ECDSA_AES_128_GCM_SHA256`) e 192 bit (por exemplo, `ECDHE_ECDSA_AES_256_GCM_SHA384`),

O diagrama a seguir ilustra o relacionamento entre estes subconjuntos:



As especificações de cifra que podem ser usadas com o suporte SSL e TLS do IBM WebSphere MQ são lisadas na tabela a seguir: Ao exigir um certificado pessoal, você especifica um tamanho de chave para o par de chaves público e particular. O tamanho de chave que é usado durante o handshake SSL

é o tamanho armazenado no certificado, a menos que ele seja determinado pelo CipherSpec, conforme indicado na tabela.

Nome do CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de criptografia	Bits de Criptografia	FIPS[1]	Conjunto B de 128 bits	Conjunto B de 192 bits
NULL_MD5 [a]	SSL 3.0	MD5	Nenhum	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Nenhum	0	No	No	No
RC4_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC4	40	No	No	No
RC4_MD5_US [a]	SSL 3.0	MD5	RC4	128	No	No	No
RC4_SHA_US [a]	SSL 3.0	SHA-1	RC4	128	No	No	No
RC2_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC2	40	No	No	No
DES_SHA_EXPORT [2] [a]	SSL 3.0	SHA-1	DES	56	No	No	No
RC4_56_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	RC4	56	No	No	No
DES_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	DES	56	No	No	No
TLS_RSA_WITH_AES_128_CBC_SHA [a]	TLS 1.0	SHA-1	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sim	No	No
TLS_RSA_WITH_DES_CBC_SHA [a]	TLS 1.0	SHA-1	DES	56	Não [5]	No	No
FIPS_WITH_DES_CBC_SHA [b]	SSL 3.0	SHA-1	DES	56	Não [6]	No	No
TLS_RSA_WITH_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sim	No	No
ECDHE_ECDSA_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No
ECDHE_RSA_RC4_128_SHA256 [b]	TLS 1.2	SHA_1	RC4	128	No	No	No
ECDHE_ECDSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sim	No	No
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sim	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	Sim	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	Sim

Nome do CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de criptografia	Bits de Criptografia	FIPS[1]	Conjunto B de 128 bits	Conjunto B de 192 bits
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Nenhum	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nenhum	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nenhum	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Nenhum	Nenhum	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Notas:

1. Especifica se o CipherSpec é certificado por FIPS em uma plataforma certificada por FIPS. Consulte Federal Information Processing Standards (FIPS) para obter uma explicação do FIPS.
2. O tamanho de chave de handshake máximo é 512 bits. Caso nenhum dos certificados trocados durante o protocolo de reconhecimento do SSL tenha um tamanho de chave superior a 512 bits, uma chave de 512 bits temporária será gerada para uso durante o protocolo de reconhecimento.
3. O tamanho de chave de handshake é 1024 bits.
4. Esse CipherSpec não pode ser usado para proteger uma conexão do WebSphere MQ Explorer com um gerenciador de filas, a menos que os arquivos de políticas irrestritos apropriados sejam aplicados ao JRE usado pelo Explorer.
5. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007.
6. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007. O nome do FIPS_WITH_DES_CBC_SHA é histórico e reflete o fato de que esse CipherSpec era anteriormente (mas não é mais) compatível com FIPS. Esse CipherSpec foi descontinuado e seu uso não é recomendado.
7. O CipherSpec pode ser usado para transferir até 32 GB de dados antes de a conexão ser finalizada com o erro AMQ9288. Para evitar esse erro, evite o uso de DES triplo ou ative a reconfiguração de chave secreta ao usar esse CipherSpec.

Suporte da Plataforma:

- a Disponível em todas as plataformas suportadas
- b Disponível apenas em plataformas UNIX, Linux, and Windows

Conceitos relacionados

"Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ" na página 35
Este tópico fornece informações sobre como escolher os certificados digitais e do CipherSpecs para sua política de segurança, delineando o relacionamento entre os certificados digitais e de CipherSpecs no IBM WebSphere MQ.

Referências relacionadas

DEFINE CHANNEL

ALTER CHANNEL

CipherSpecs descontinuado

Uma lista de CipherSpecs descontinuados que você pode usar com o WebSphere MQ , se necessário.

Consulte “CipherSpec valores suportados em IBM WebSphere MQ” na página 39 para obter mais informações sobre como é possível ativar os CipherSpecs descontinuados.

Os CipherSpecs descontinuados que podem ser usados com o suporte TLS do WebSphere MQ são listados na tabela a seguir:

Suporte da plataforma “1” na página 226	Nome do CipherSpec	Protocolo utilizado	Integridade de dados	Algoritmo de criptografia	Bits de Criptografia	FIPS “2” na página 226	Conjunto B	Atualizar quando descontinuado
Todos(as)	DES_SHA_EXPORT “3” na página 226	SSL 3.0	SHA-1	Padrão de Criptografia de Dados	56	Não	Não	7.5.0.6
Windows UNIX Linux	DES_SHA_EXPORT1024 “4” na página 226	SSL 3.0	SHA-1	Padrão de Criptografia de Dados	56	Não	Não	7.5.0.6
Windows UNIX Linux	FIPS_WITH_DES_CBC_SHA	SSL 3.0	SHA-1	Padrão de Criptografia de Dados	56	Não “6” na página 226	Não	7.5.0.6
Windows UNIX Linux	FIPS_WITH_3DES_EDE_CBC_SHA	SSL 3.0	SHA-1	3DES	168	Não “7” na página 226	Não	7.5.0.8
Todos(as)	NULL_MD5	SSL 3.0	MD5	Nenhum	0	Não	Não	7.5.0.6
Todos(as)	NULL_SHA	SSL 3.0	SHA-1	Nenhum	0	Não	Não	7.5.0.6
Todos(as)	RC2_MD5_EXPORT “3” na página 226	SSL 3.0	MD5	RC2	40	Não	Não	7.5.0.7
Todos(as)	RC4_MD5_EXPORT “3” na página 226	SSL 3.0	MD5	RC4	40	Não	Não	7.5.0.7
Todos(as)	RC4_MD5_US	SSL 3.0	MD5	RC4	128	Não	Não	7.5.0.7
Todos(as)	RC4_SHA_US	SSL 3.0	SHA-1	RC4	128	Não	Não	7.5.0.7
Windows UNIX Linux	RC4_56_SHA_EXPORT1024 “4” na página 226	SSL 3.0	SHA-1	RC4	56	Não	Não	7.5.0.7

Suporte da plataforma "1" na página 226	Nome do CipherSpec	Protocolo utilizado	Integridade de dados	Algoritmo de criptografia	Bits de Criptografia	FIPS "2" na página 226	Conjunto B	Atualizar quando descontinuado
Todos(as)	TRIPLE_DES_SHA_US	SSL 3.0	SHA-1	3DES	168	Não	Não	7.5.0.8
Todos(as)	TLS_RSA_WITH_DES_CBC_SHA	TLS 1.0	SHA-1	Padrão de Criptografia de Dados	56	Não"5" na página 226	Não	7.5.0.6
Windows UNIX Linux	ECDHE_ECDSA_NULL_SHA256	TLS 1.2	SHA-1	Nenhum	0	Não	Não	7.5.0.6
Windows UNIX Linux	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	Não	Não	7.5.0.7
Windows UNIX Linux	ECDHE_RSA_NULL_SHA256	TLS 1.2	SHA-1	Nenhum	0	Não	Não	7.5.0.6
Windows UNIX Linux	ECDHE_RSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	Não	Não	7.5.0.7
Windows UNIX Linux	TLS_RSA_WITH_NULL_NULL	TLS 1.2	Nenhum	Nenhum	0	Não	Não	7.5.0.6
Todos(as)	TLS_RSA_WITH_NULL_SHA256	TLS 1.2	SHA-256	Nenhum	0	Não	Não	7.5.0.6
Windows UNIX Linux	TLS_RSA_WITH_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	Não	Não	7.5.0.7
Todos(as)	TLS_RSA_WITH_3DES_EDE_CBC_SHA"8" na página 226	TLS 1.0	SHA-1	3DES	168	Sim	Não	7.5.0.8
Windows UNIX Linux	ECDHE_ECDSA_3DES_EDE_CBC_SHA256"8" na página 226	TLS 1.2	SHA-1	3DES	168	Sim	Não	7.5.0.8
Windows UNIX Linux	ECDHE_RSA_3DES_EDE_CBC_SHA256"8" na página 226	TLS 1.2	SHA-1	3DES	168	Sim	Não	7.5.0.8

Suporte da plataforma "1" na página 226	Nome do CipherSpec	Protocolo utilizado	Integridade de dados	Algoritmo de criptografia	Bits de Criptografia	FIPS "2" na página 226	Conjunto B	Atualizar quando descontinuado

Notes:

1. Se nenhuma plataforma específica for observada, o CipherSpec estará disponível em todas as plataformas.
2. Especifica se o CipherSpec é certificado por FIPS em uma plataforma certificada por FIPS. Consulte Federal Information Processing Standards (FIPS) para obter uma explicação do FIPS.
3. O tamanho de chave de handshake máximo é 512 bits. Caso nenhum dos certificados trocados durante o protocolo de reconhecimento do SSL tenha um tamanho de chave superior a 512 bits, uma chave de 512 bits temporária será gerada para uso durante o protocolo de reconhecimento.
4. O tamanho de chave de handshake é 1024 bits.
5. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007.
6. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007. O nome `FIPS_WITH_DES_CBC_SHA` é histórico e reflete o fato de que este CipherSpec era anteriormente (mas não é mais) compatível com FIPS. Esse CipherSpec foi descontinuado e seu uso não é recomendado.
7. O nome `FIPS_WITH_3DES_EDE_CBC_SHA` é histórico e reflete o fato de que este CipherSpec era anteriormente (mas não é mais) compatível com FIPS. O uso deste CipherSpec está descontinuado.
8. O CipherSpec pode ser usado para transferir até 32 GB de dados antes de a conexão ser finalizada com o erro AMQ9288. Para evitar esse erro, evite o uso de DES triplo ou ative a reconfiguração de chave secreta ao usar esse CipherSpec.

Obtendo informações sobre CipherSpecs usando o IBM WebSphere MQ Explorer

É possível usar o IBM WebSphere MQ Explorer para exibir descrições de CipherSpecs.

Utilize o seguinte procedimento para obter informações sobre o CipherSpecs no "Especificando CipherSpecs" na página 221:

1. Abra o **IBM WebSphere MQ Explorer** e expanda a pasta **Gerenciadores de filas**.
2. Certifique-se de que iniciou o gerenciador de filas.
3. Selecione o Gerenciador de Filas com o qual deseja trabalhar e clique em **Canais**.
4. Clique com o botão direito no canal que deseja trabalhar e selecione **Propriedades**.
5. Selecione a página de propriedades **SSL**.
6. Selecione da lista o CipherSpec com o qual quer trabalhar. Uma descrição é exibida na janela abaixo da lista.

Alternativas para a Especificação do CipherSpecs

Para as plataformas em que o sistema operacional fornece suporte ao SSL, é possível que o sistema suporte novos CipherSpecs. Você pode especificar um novo CipherSpec com o parâmetro SSLCIPH, mas o valor que você fornecer dependerá da sua plataforma.

Nota: Esta seção não se aplica a sistemas UNIX, Linux ou Windows , porque os CipherSpecs são fornecidos com o produto WebSphere MQ , portanto, os novos CipherSpecs não se tornam disponíveis após a remessa

Para aquelas plataformas em que o sistema operacional fornece suporte ao SSL, é possível que o seu sistema suporte os novos CipherSpecs que não são incluídos no "Especificando CipherSpecs" na página 221. Você pode especificar um novo CipherSpec com o parâmetro SSLCIPH, mas o valor que você

fornecer dependerá da sua plataforma. Em todos os casos, a especificação *deve* corresponder a um SSL CipherSpec que seja válido e suportado pela versão do SSL com o qual seu sistema está sendo executado.

IBM i
Uma cadeia de dois caracteres que representa um valor hexadecimal.

Para obter mais informações sobre os valores permitidos, consulte a documentação apropriada do produto (procure *cipher_spec* na documentação do produtoIBM i).

Você pode utilizar o comando CHGMQMCHL ou CRTMQMCHL para especificar o valor, por exemplo:

```
CRTMQMCHL CHLNAME('channel name') SSLCIPH('hexadecimal value')
```

Também é possível usar o comando ALTER QMGR MQSC para configurar o parâmetro SSLCIPH.

z/OS
Uma cadeia de dois caracteres que representa um valor hexadecimal. Os códigos hexadecimais correspondem aos valores definidos no protocolo SSL.

Para obter mais informações, consulte a descrição de gsk_environment_open () no capítulo de referência da API do *z/OS Cryptographic Services System SSL Programming*, SC24-5901, em que há uma lista de todas as especificações de código SSL V3.0 e TLS V1.0 suportadas no formato de códigos hexadecimais de 2 dígitos.

Considerações para clusters do WebSphere MQ

Com clusters do WebSphere MQ , é mais seguro usar os nomes CipherSpec em “Especificando CipherSpecs” na página 221. Se você usar uma especificação alternativa, tenha em mente que a especificação pode não ser válida em outras plataformas. Para obter informações adicionais, consulte “SSL e clusters” na página 253.

Especificando um CipherSpec para um cliente MQI do IBM WebSphere MQ

Você tem três opções para especificar um CipherSpec para um cliente MQI do IBM WebSphere MQ

Estas opções são as seguintes:

- Utilizando uma tabela de definição de canais
- Usando o campo SSLCipherSpec na estrutura MQCD, no MQCD_VERSION_7 ou mais recente, em uma chamada MQCONNX.
- Usando o Active Directory (em sistemas Windows com suporte do Active Directory)

Especificando um CipherSuite com classes IBM WebSphere MQ para Java e classes IBM WebSphere MQ para JMS

As classes IBM WebSphere MQ para Java e IBM WebSphere MQ para JMS especificam CipherSuites de forma diferente de outras plataformas.

Para obter informações sobre como especificar um CipherSuite com IBM WebSphere MQ classes para Java, consulte Suporte SSL (Secure Sockets Layer).

Para obter informações sobre como especificar um CipherSuite com IBM WebSphere MQ classes para JMS, consulte Usando Secure Sockets Layer (SSL) com WebSphere MQ classes para JMS.

Reconfigurando as chaves secretas SSL e TLS

O IBM WebSphere MQ suporta a reinicialização de chaves secretas em gerenciadores de filas e clientes.

As chaves secretas são reconfiguradas quando um número especificado de bytes criptografados de dados é transmitido pelo canal ou depois que o canal fica inativo por um período.

O valor de reconfiguração de chave é sempre configurado pelo lado inicial do canal do MQ .

Gerenciador de Filas

Para um gerenciador de filas, use o comando **ALTER QMGR** com o parâmetro **SSLRKEYC** para definir os valores usados durante a renegociação.

Cliente MQI

Por padrão, os clientes MQI não renegociam a chave secreta. É possível fazer um cliente MQI renegociar a chave em qualquer uma das três formas. Na lista a seguir, os métodos são mostrados em ordem de prioridade. Se você especificar diversos valores, o valor de prioridade mais alto será usado.

1. Usando o campo KeyResetCount na estrutura MQSCO em uma chamada MQCONNX
2. Usando a variável de ambiente MQSSLRESET
3. Configurando o atributo SSLKeyResetCount no arquivo de configuração do cliente MQI

Estas variáveis podem ser configuradas como um número inteiro no intervalo de 0 até 999999999, representando o número de bytes não criptografados enviados e recebidos em uma conversação SSL ou TLS antes da chave secreta SSL ou TLS for renegociada. Especificar um valor igual a 0 indica que as chaves secretas SSL ou TLS nunca serão renegociadas. Se você especificar uma contagem de reconfiguração de chave secreta SSL ou TLS no intervalo de 1 byte até 32 KB, os canais SSL ou TLS usarão uma contagem de reconfiguração de chave secreta de 32 KB. Isto é para evitar reconfigurações de chave excessivas que ocorreriam para valores pequenos de reconfiguração de chave secreta SSL ou TLS.

Se um valor maior que zero for especificado e as pulsações de canal forem ativadas para o canal, a chave secreta também será renegociada antes dos dados da mensagem serem enviados ou recebidos após uma pulsação de canal.

A contagem de bytes até a próxima renegociação de chave secreta é reconfigurada após cada renegociação bem-sucedida.

Para obter detalhes integrais da estrutura MQSCO, veja KeyResetCount (MQLONG). Para obter detalhes completos de MQSSLRESET, consulte MQSSLRESET. Para obter mais informações sobre o uso de SSL ou TLS no arquivo de configuração do cliente, consulte Sub-rotina de SSL do arquivo de configuração do cliente.

Java

Para o IBM WebSphere MQ classes for Java, um aplicativo pode reconfigurar a chave secreta de uma das maneiras a seguir:

- Configurando o campo sslResetCount na classe MQEnvironment.
- Configurando uma propriedade de ambiente MQC.SSL_RESET_COUNT_PROPERTY em um objeto Hashtable. O aplicativo designa, então, a hashtable para o campo `properties` na classe MQEnvironment ou passa a hashtable para um objeto MQQueueManager em seu construtor.

Se o aplicativo usar mais de uma dessas maneiras, as regras usuais de precedência se aplicam. Consulte Classe com.ibm.mq.MQEnvironment para as regras de precedência.

O valor do campo de contagem sslResetou da propriedade de ambiente MQC.SSL_RESET_COUNT_PROPERTY representa o número total de bytes enviados e recebidos pelas classes WebSphere MQ para o código do cliente Java antes que a chave secreta seja renegociada. O número de bytes enviados é o número antes da criptografia e o número de bytes recebidos é o número após a decriptografia. O número de bytes também inclui informações de controle enviadas e recebidas pelas classes do WebSphere MQ para cliente Java.

Se a contagem de reconfiguração for zero, que é o valor padrão, a chave secreta nunca será renegociada. A contagem de reconfiguração será ignorada se nenhum CipherSuite for especificado.

JMS

Para o IBM WebSphere MQ classes for JMS, a propriedade SSLRESETCOUNT representa o número total de bytes enviados e recebidos por uma conexão antes que chave secreta que é usada para criptografia seja renegociada. O número de bytes enviados é o número antes da criptografia e o número de bytes recebidos é o número após a decriptografia. O número de bytes também inclui informações de controle enviadas e recebidas pelo IBM WebSphere MQ classes for JMS. Por exemplo, para configurar um objeto ConnectionFactory que possa ser usado para criar uma conexão sobre um canal MQI ativado para SSL ou TLS com uma chave secreta renegociada após a transmissão de 4 MB de dados, emita o comando a seguir para JMSAdmin:

```
ALTER CF(my.cf) SSLRESETCOUNT(4194304)
```

Se o valor de SSLRESETCOUNT for zero, que é o valor padrão, a chave secreta nunca será renegociada. A propriedade SSLRESETCOUNT será ignorada se SSLCIPHERSUITE não estiver configurado.

.REDE

Para clientes não gerenciados .NET, a contagem de propriedade de número inteiro SSLKeyResetindica o número de bytes não criptografados enviados e recebidos em uma conversa SSL ou TLS antes que a chave secreta seja renegociada.

Para obter informações sobre o uso de propriedades de objeto em classes IBM WebSphere MQ para .NET, consulte Obtendo e configurando valores de atributo.

XMS .NET

Para clientes não gerenciados do XMS .NET, consulte Conexões seguras para um IBM WebSphere MQ gerenciador de filas

Referências relacionadas
ALTER QMGR
DISPLAY QMGR

Implementando confidencialidade em programas de saída do usuário

Implementando confidencialidade em saídas de segurança

As saídas de segurança podem exercer uma função no serviço de confidencialidade gerando e distribuindo a chave simétrica para criptografia e decriptografia de dados que flui no canal. Uma técnica comum aplicada utiliza a tecnologia PKI.

Uma saída de segurança gera um valor de dados aleatório, criptografa-os com a tecla pública do gerenciador ou usuário de fila que a saída do parceiro está representando e envia os dados criptografados a seu parceiro em uma mensagem de segurança. A saída de segurança do parceiro decriptografa os valores de dados aleatórios com a tecla privativa do gerenciador ou usuário de fila que os está representando. Cada saída de segurança pode agora utilizar o valor de dados aleatórios para derivar a chave simétrica, independentemente de outros utilizando um algoritmo conhecido de ambos. De forma alternativa, podem usar o valor de dados aleatórios como a chave.

Se a primeira saída de segurança não autenticou seu parceiro até então, a mensagem de segurança seguinte enviada pelo parceiro poderá conter uma valor inesperado criptografado com a chave simétrica. A primeira saída de segurança pode então autenticar seu parceiro verificando se a saída de segurança dele conseguiu criptografar o valor esperado corretamente.

As saídas de segurança podem também usar esta oportunidade para concordar o algoritmo para criptografia e decriptografia de dados que fluem no canal, se mais de um algoritmo estiver disponível para uso.

Implementando confidencialidade em saídas de mensagem

Uma saída de mensagem na extremidade de envio de um canal pode criptografar os dados do aplicativo em uma mensagem e outra saída na extremidade de recepção do canal pode descriptografar os dados. Por razões de desempenho, um algoritmo de tecla simétrica é normalmente utilizado para este propósito. Para obter mais informações sobre como a chave simétrica pode ser gerada e distribuída, consulte “Implementando confidencialidade em programas de saída do usuário” na página 229.

Os cabeçalhos de uma mensagem, como da fila de transmissão, MQXQH, que inclui o descritor da mensagem interna, não devem ser criptografados por uma saída de mensagem. Isso porque a conversão de dados dos cabeçalhos da mensagem ocorre depois que uma saída de mensagem é chamada na extremidade de envio ou antes de ser chamada na extremidade de recepção. Se os cabeçalhos forem criptografados, a conversão de dados falhará e o canal parará.

Implementando confidencialidade em saídas de envio e de recebimento

As saídas de envio e recebimento podem ser utilizadas para criptografar e descriptografar dados que passam por um canal. Elas são mais apropriadas que as saídas de mensagens que fornecem esse serviço pelos seguintes motivos:

- Em um canal de mensagem, os cabeçalhos da mensagem podem ser criptografados assim como os dados do aplicativo nas mensagens.
- As saídas de envio e recebimento podem ser usadas em canais MQI assim como em canais de mensagens. Os parâmetros nas chamadas MQI podem não conter dados sensíveis do aplicativo que tenham que ser protegidos enquanto passam em um canal MQI. Portanto, é possível usar as mesmas saídas de envio e recebimento nos dois tipos de canais.

A confidencialidade na saída da API e saída cruzada da API

Os dados do aplicativo em uma mensagem podem ser criptografados por uma saída API ou saída cruzada da API quando a mensagem for colocada pelo aplicativo de envio e decriptografada por uma segunda saída quando a mensagem for recuperada pelo aplicativo de recebimento. Por razões de desempenho, um algoritmo de chave simétrica é normalmente usado para este propósito. No entanto, ao nível do aplicativo, em que muitos usuários podem estar enviando mensagens uns aos outros, o problema é como assegurar que somente o receptor pretendido de uma mensagem seja capaz de decriptografá-la. Uma solução é utilizar uma chave simétrica diferente para cada par de usuários que enviem mensagens uns aos outros. Mas essa solução pode ser difícil e demorada para administrar, particularmente se os usuários pertencerem a organizações diferentes. Um modo padrão de resolver este problema é conhecido como *envelopamento digital* e utiliza tecnologia PKI.

Quando um aplicativo coloca uma mensagem em uma fila, uma saída API ou saída cruzada da API gera uma chave simétrica aleatória e usa a chave para criptografar os dados do aplicativo na mensagem. A saída criptografa a chave simétrica com a chave pública do receptor desejado. Então, ela substitui os dados do aplicativo na mensagem pelos dados criptografados do aplicativo e a chave simétrica criptografada. Deste modo, somente o receptor pretendido poderá decriptografar a chave simétrica e, portanto, os dados do aplicativo. Se uma mensagem criptografada tiver mais de um possível receptor desejado, a saída poderá criptografar uma cópia da chave simétrica para cada receptor desejado.

Se diferentes algoritmos para criptografar e decriptografar dados do aplicativo estiverem disponíveis para uso, a saída poderá incluir o nome do algoritmo que foi usado.

Integridade de dados de mensagens

Para manter a integridade dos dados é possível usar vários tipos de programa de saída de usuário para fornecer trechos de mensagens ou assinaturas digitais para suas mensagens.

Integridade de dados

Implementando integridade de dados em mensagens

Quando você usar SSL ou TLS, sua opção de CipherSpec determina o nível de integridade de dados na empresa. Se você usar o WebSphere MQ Advanced Message Service (AMS), poderá especificar a integridade para uma mensagem exclusiva.

Implementando integridade de dados em saídas de mensagem

Uma mensagem pode ser assinada digitalmente por uma saída de usuário na extremidade de envio de um canal. A assinatura digital pode então ser verificada por uma saída de mensagem na extremidade de recepção de um canal para detectar se a mensagem foi modificada deliberadamente.

Alguma proteção pode ser proporcionada utilizando-se uma compilação de mensagens ao invés de uma assinatura digital. Uma compilação de mensagens pode ser efetiva contra violação casual ou indiscriminada, mas não evita que pessoas mais informadas alterem ou substituam a mensagem e gerem uma compilação completamente nova. Isto é particularmente verdadeiro quando o algoritmo utilizado para gerar a compilação da mensagem é bem conhecido.

Implementando integridade de dados em saídas de envio e recebimento

Em um canal de mensagem, as saídas de mensagem são mais apropriadas para fornecer esse serviço porque uma saída pode acessar uma mensagem inteira. Em um canal MQI, os parâmetros nas chamadas MQI podem conter dados do aplicativo que precisem de proteção e somente as saídas de envio e recebimento podem fornecer essa proteção.

Implementando integridade de dados na saída da API ou saída cruzada da API

Uma mensagem pode ser assinada digitalmente por uma saída API ou saída cruzada da API ao ser colocada pelo aplicativo de envio. A assinatura digital pode então ser verificada por uma segunda saída quando a mensagem for recuperada pelo aplicativo receptor para detectar se a mensagem foi modificada deliberadamente.

Alguma proteção pode ser proporcionada utilizando-se uma compilação de mensagens ao invés de uma assinatura digital. Uma compilação de mensagens pode ser efetiva contra violação casual ou indiscriminada, mas não evita que pessoas mais informadas alterem ou substituam a mensagem e gerem uma compilação completamente nova. Isso é particularmente verdadeiro se o algoritmo que é usado para gerar o trecho da mensagem é conhecido,

Conectando dois gerenciadores de filas usando SSL ou TLS

As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

Para configurar sua instalação de SSL ou TLS, você deve definir seus canais para usar SSL ou TLS. Você também deve obter e gerenciar seus certificados digitais. Em um sistema de teste, é possível usar certificados autoassinados ou certificados emitidos por uma autoridade de certificação (CA) local. Em um sistema de produção, não use certificados autoassinados. Para obter mais informações, consulte ../zs14140_.dita.

Para obter informações completas sobre como criar e gerenciar certificados, consulte “Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows” na página 116.

Esta coleção de tópicos apresenta as tarefas envolvidas na configuração de comunicações SSL e fornece orientação passo a passo sobre como concluir as tarefas.

Você também pode querer testar a autenticação de cliente SSL ou TLS, que são uma parte opcional dos protocolos. Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação do WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente.

Notes:

1. Neste contexto, um cliente SSL refere-se à conexão que está iniciando o handshake.
2. Consulte o Glossário para obter detalhes adicionais

Nos sistemas UNIX, Linux e Windows , o cliente SSL ou TLS envia um certificado somente se ele tiver um rotulado no formato correto do WebSphere MQ , que é `ibmwebspheremq` seguido pelo nome do seu gerenciador de filas alterado para minúsculas. Por exemplo, para `QM1`, `ibmwebspheremqqm1`.

WebSphere MQ usa o prefixo `ibmwebspheremq` em um rótulo para evitar confusão com certificados para outros produtos. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falha se a extremidade do canal que está agindo como o servidor SSL ou TLS estiver definida com o parâmetro SSLCAUTH configurado para REQUIRED ou um conjunto de valores do parâmetro SSLPEER. Para obter informações adicionais sobre como conectar um gerenciador de filas de forma anônima ou seja, quando o cliente SSL ou TLS não enviar um certificado, consulte “Conectando dois gerenciadores de filas usando autenticação unilateral” na página 214.

Rótulos de Certificados Digitais, Entendendo os Requisitos

Ao configurar o SSL e o TLS para usar certificados digitais, pode haver requisitos de rótulo específicos que devem ser seguidos, dependendo da plataforma usada e do método usado para a conexão.

Sobre esta tarefa

O que é rótulo certificado?

Um rótulo de certificado é um identificador exclusivo que representa um certificado digital armazenado em um repositório de chaves e que fornece um nome legível conveniente com o qual se referir a um determinado certificado ao executar funções de gerenciamento de chaves. Você designa o rótulo certificado ao incluir um certificado em um repositório de chaves pela primeira vez.

O rótulo certificado é separado dos campos *Subject Distinguished Name* ou *Subject Common Name* do certificado. Observe que o *Nome Distinto do Assunto* e o *Nome Comum do Assunto* são campos dentro do próprio certificado. Eles são definidos quando o certificado é criado e não pode ser alterado. No entanto, é possível alterar o rótulo associado a um certificado digital, se necessário,.

Como o rótulo certificado é usado?

O IBM WebSphere MQ usa rótulos de certificado para localizar um certificado pessoal que é enviado durante o handshake SSL. Isso elimina a ambiguidade quando existe mais de um certificado pessoal no repositório de chaves.

Os rótulos de certificado seguem uma convenção de nomenclatura; você precisa assegurar que use a convenção de nomenclatura de rótulo correta correspondente à plataforma que está sendo usada.

Nesse contexto, um cliente SSL ou TLS refere-se ao parceiro de conexão que inicia o handshake, que pode ser um cliente IBM WebSphere MQ ou outro gerenciador de filas.

Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação IBM WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente e o cliente sempre fornece um certificado para o servidor se um for localizado. Se o cliente não puder localizar um certificado pessoal, o cliente enviará uma resposta no `certificate` para o servidor

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falhará se a extremidade do canal que está agindo como o servidor SSL ou TLS for definida com o parâmetro SSLCAUTH configurado como REQUIRED ou um valor de parâmetro SSLPEER configurado.

Para obter mais informações sobre como conectar um gerenciador de filas usando autenticação unilateral, ou seja, quando o cliente SSL ou TLS não envia um certificado, consulte “Conectando dois gerenciadores de filas usando autenticação unilateral” na página 214.

Sistemas , UNIX, Linux, and Windows

Sobre esta tarefa

Nos sistemas , UNIX, Linux, and Windows , o servidor SSL ou TLS envia um certificado para o cliente, somente se o servidor localizar um rotulado no formato IBM WebSphere MQ correto Nesses sistemas, o formato correto é `ibmwebspheremq`, seguido pelo nome de seu gerenciador de filas alterado para minúsculas.

Por exemplo, para um gerenciador de filas denominado `QM1`, o requisito do rótulo certificado é:

```
ibmwebspheremqqm1
```

Se nenhum certificado for localizado no repositório de chaves do gerenciador de filas, correspondendo ao rótulo necessário no caso e no formato corretos, ocorrerá um erro e o handshake SSL ou TLS falhará

IBM WebSphere MQ client

Sobre esta tarefa

Ao se conectar a partir de um aplicativo cliente IBM WebSphere MQ , o cliente SSL ou TLS envia um certificado apenas se ele tiver um certificado com um rótulo no formato `ibmwebspheremq`, seguido pelo nome do usuário do usuário que está executando o processo de aplicativo cliente.

Por exemplo, para o nome do usuário `wasadmin`, o requisito do rótulo certificado é conforme mostrado, dobrado para minúsculas:

```
  ibmwebspheremqwasadmin
```

O requisito de rótulo acima se aplica aos Clientes de Serviço de Mensagens para C ou C + + e .NET.

Cliente JMS IBM WebSphere MQ Java ou IBM WebSphere MQ

Sobre esta tarefa

IBM WebSphere MQ Os clientes Java ou IBM WebSphere MQ JMS usam os recursos de seu provedor Java Secure Socket Extension (JSSE) para selecionar um certificado pessoal durante o handshake SSL ou TLS e, portanto, não estão sujeitos aos requisitos de rótulo certificado.

O comportamento padrão é que o cliente JSSE itera por meio dos certificados no repositório de chaves, selecionando o primeiro certificado pessoal aceitável localizado No entanto, esse comportamento é apenas um padrão, e depende da implementação do provedor JSSE.

Além disso, a interface JSSE é altamente customizável por meio de configuração e acesso direto no tempo de execução pelo aplicativo. Consulte a documentação fornecida pelo provedor JSSE para obter detalhes específicos.

Para resolução de problemas ou para entender melhor o handshake executado pelo aplicativo cliente Java do IBM WebSphere MQ em combinação com seu provedor JSSE específico, é possível ativar a depuração configurando

```
javax.net.debug=ssl
```

no ambiente JVM.

É possível usar `-Djavax.net.debug=ssl` na linha de comandos ou configurar a variável dentro do aplicativo ou por meio da configuração

Conceitos relacionados

“Importando um certificado pessoal em um repositório de chaves em sistemas UNIX, Linux, and Windows” na página 136
Siga este procedimento para importar um certificado pessoal

Usando certificados autoassinados para autenticação mútua de dois gerenciadores de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre dois gerenciadores de filas, usando certificados SSL ou TLS autoassinados.

Sobre esta tarefa

Cenário:

- Você possui dois gerenciadores de filas, QM1 e QM2, que precisam se comunicar com segurança. É necessário que a autenticação mútua seja executada entre QM1 e QM2.
- Você decidiu testar a comunicação segura usando certificados autoassinados.

A configuração resultante é semelhante à seguinte:

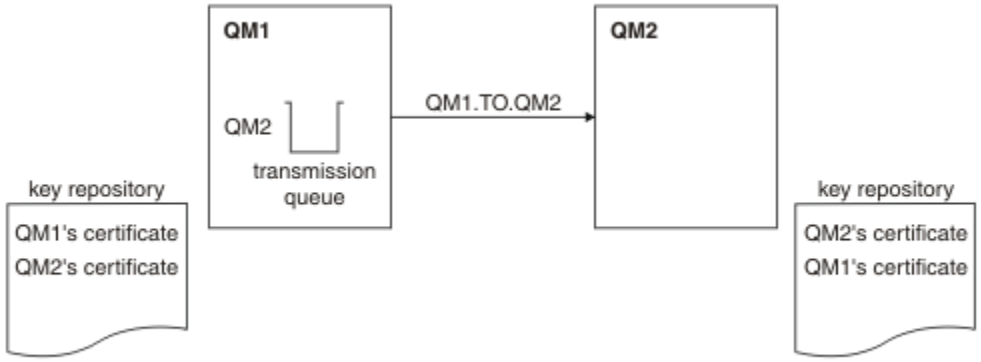


Figura 20. Configuração resultante desta tarefa

Na Figura 14 na página 211, o repositório de chaves para QM1 contém os certificados para QM1 e o certificado público de QM2. O repositório de chaves para QM2 contém o certificado para QM2 e o certificado público de QM1.

Procedimento

1. Prepare o repositório de chaves em cada gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Crie um certificado autoassinado para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
3. Extraia uma cópia de cada certificado:
   - Em UNIX, Linuxe Windows sistemas
4. Transfira a parte pública do certificado QM1 para o sistema QM2 e vice-versa, usando um utilitário como FTP.
5. Inclua o certificado do parceiro no repositório de chaves para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
6. No QM1, defina um canal emissor e a fila de transmissão associada, emitindo os comandos como o seguinte exemplo:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(RC4_MD5_US) DESCR('Sender channel using SSL from QM1 to QM2')

DEFINE QLOCAL(QM2) USAGE(XMITQ)
```

Este exemplo usa CipherSpec RC4_MD5. Os CipherSpecs em cada extremidade do canal devem ser iguais.

7. No QM2, defina um canal receptor, emitindo um comando como o seguinte exemplo:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from QM1 to QM2')
```

   O canal deve ter o mesmo nome que o canal emissor definido na etapa 6 e usar o mesmo CipherSpec.

8. Inicie o canal.

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 14 na página 211

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM1.TO.QM2)                     CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=QM2,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

No gerenciador de filas QM2, insira o seguinte comando:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM2.TO.QM1)                     CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QM1)
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

Em cada caso, o valor de SSLPEER deve corresponder àquele do DN no certificado parceiro que foi criado na Etapa 2. O nome do emissor corresponde ao nome do peer porque o certificado é autoassinado.

SSLPEER é opcional. Se for especificado, o valor deve ser configurado para que o DN no certificado do parceiro (criado na etapa 2) seja permitido. Para obter mais informações sobre o uso de SSLPEER, consulte WebSphere MQ regras para valores SSLPEER .

Usando certificados assinados por CA para autenticação mútua de dois gerenciadores de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre dois gerenciadores de filas, usando certificados SSL ou TLS assinados por CA.

Sobre esta tarefa

Cenário:

- Você possui dois gerenciadores de filas, QMA e QMB, que precisam se comunicar com segurança. É necessário executar a autenticação mútua entre QMA e QMB.
- No futuro, você está planejando usar essa rede em um ambiente de produção e, portanto, decidiu usar os certificados assinados por CA desde o início.

A configuração resultante é semelhante à seguinte:

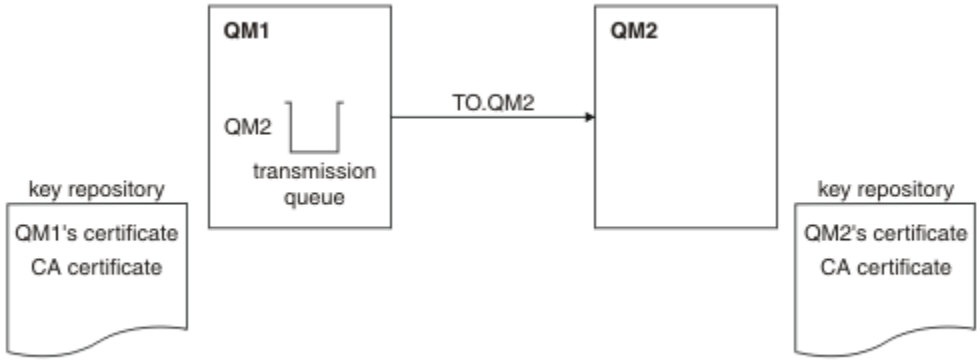


Figura 21. Configuração resultante desta tarefa

Na Figura 15 na página 213, o repositório de chaves para QMA contém o certificado de QMA e o certificado de CA. O repositório de chaves para QMB contém o certificado de QMB e o certificado de CA. Neste exemplo, o certificado de QMA e o certificado de QMB foram emitidos pela mesma CA. Se o certificado de QMA e o certificado de QMB foram emitidos por CAs diferentes, os repositórios de chaves para QMA e QMB deverão conter ambos os certificados de CA.

Procedimento

1. Prepare o repositório de chaves em cada gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Solicite um certificado assinado por CA para cada gerenciador de filas.

   É possível usar CAs diferentes para os dois gerenciadores de filas.
   - Em UNIX, Linuxe Windows sistemas
3. Inclua o certificado de Autoridade de certificação no repositório de chaves para cada gerenciador de filas:

   Se os Gerenciadores de filas estiverem usando Autoridades de certificação diferentes, o certificado de CA para cada Autoridade de certificação deverá ser incluído nos dois repositórios de chaves.
   - Em UNIX, Linuxe Windows sistemas
4. Incluir o certificado assinado por CA no repositório de chaves para cada gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
5. No QMA, defina um canal emissor e a fila de transmissão associada emitindo comandos como o seguinte exemplo:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
SSLCIPH(RC2_MD5_EXPORT) DESCR('Sender channel using SSL from QMA to QMB')

DEFINE QLOCAL(QMB) USAGE(XMITQ)
```

Este exemplo usa CipherSpec RC4_MD5. Os CipherSpecs em cada extremidade do canal devem ser iguais.

6. No QMB, defina um canal receptor, emitindo um comando como o exemplo a seguir:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
```

O canal deve ter o mesmo nome que o canal emissor definido na etapa 6 e usar o mesmo CipherSpec.

7. Inicie o canal:

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 15 na página 213.

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir.

No gerenciador de filas QMA, insira o seguinte comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QMB)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QMB)
```

No gerenciador de filas QMB, insira o seguinte comando:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

Em cada caso, o valor de SSLPEER deve corresponder àquele do Nome Distinto (DN) no certificado parceiro que foi criado na Etapa 2. O nome do emissor corresponde ao DN do sujeito do certificado CA que assinou o certificado pessoal incluído na Etapa 4.

Conectando dois gerenciadores de filas usando autenticação unilateral

Siga estas instruções de amostra para modificar um sistema com autenticação mútua para permitir que um gerenciador de filas se conecte a outro usando autenticação unilateral; ou seja, quando o cliente SSL ou TLS não enviar um certificado.

Sobre esta tarefa

Cenário:

- Os dois gerenciadores de filas (QM1 e QM2) foram configurados conforme descrito em "Usando certificados assinados por CA para autenticação mútua de dois gerenciadores de filas" na página 212.
- Você deseja alterar o QM1 para que ele se conecte ao QM2 usando autenticação unilateral.

A configuração resultante é semelhante à seguinte:

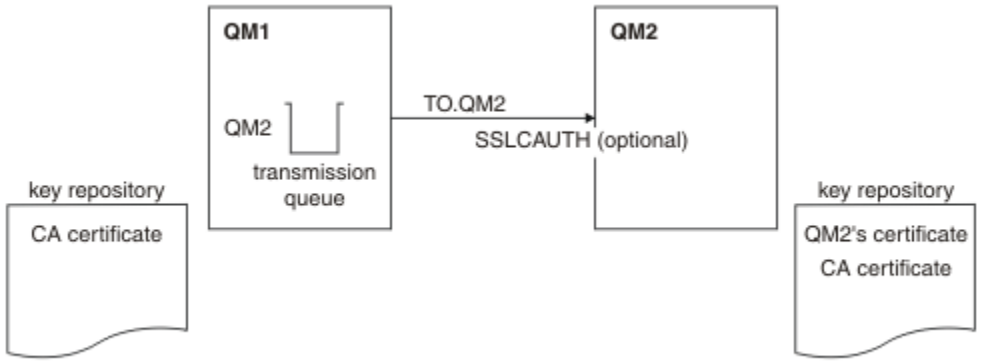


Figura 22. Gerenciadores de filas que permitem autenticação unilateral

Procedimento

1. Remova o certificado pessoal do QM1de seu repositório de chaves, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas O certificado é rotulado como a seguir:
     - `ibmwebspheremq` seguido pelo nome de seu gerenciador de filas dobrado para letras minúsculas Por exemplo, para `QM1` , `ibmwebspheremqqm1`..
2. Opcional: Em QM1, se qualquer canal SSL ou TLS tiver sido executado anteriormente, atualize o ambiente SSL ou TLS.
3. Permita conexões anônimas no

Resultados

Os repositórios de chaves e canais são mudados conforme ilustrado na Figura 16 na página 215

Como proceder a seguir

Se o canal emissor estava em execução e você emitiu o comando REFRESH SECURITY TYPE(SSL) (na etapa 2), o canal será reiniciado automaticamente. Se o canal emissor não estiver em execução, inicie-o.

Na extremidade do canal do servidor, a presença do valor de parâmetro do nome do peer na exibição de status do canal indica que um certificado de cliente foi emitido.

Verifique se a tarefa foi concluída com sucesso, emitindo alguns comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada nos exemplos a seguir:

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

No gerenciador de filas QM2, insira o seguinte comando:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)                            SSLCERTI( )
   SSLPEER( )                              STATUS(RUNNING)
   SUBSTATE(RECEIVE)                       XMITQ( )
```

No QM2, o campo SSLPEER está vazio, mostrando que o QM1 não enviou um certificado. No QM1, o valor de SSLPEER corresponde ao valor do DN no certificado pessoal do QM2.

Conectando um Cliente a um Gerenciador de Filas de Forma Segura

As comunicações seguras que usam os protocolos de segurança criptográfica SSL ou TLS envolvem a configuração dos canais de comunicação e o gerenciamento de certificados digitais que serão usados para autenticação.

Para configurar sua instalação de SSL ou TLS, você deve definir seus canais para usar SSL ou TLS. Você também deve obter e gerenciar seus certificados digitais. Em um sistema de teste, é possível usar certificados autoassinados ou certificados emitidos por uma autoridade de certificação (CA) local. Em um sistema de produção, não use certificados autoassinados. Para obter mais informações, consulte ../zs14140_.dita.

Para obter informações completas sobre como criar e gerenciar certificados, consulte “Trabalhando com SSL ou TLS em sistemas UNIX, Linux, and Windows” na página 116.

Esta coleção de tópicos apresenta as tarefas envolvidas na configuração de comunicações SSL e fornece orientação passo a passo sobre como concluir as tarefas.

Você também pode querer testar a autenticação de cliente SSL ou TLS, que são uma parte opcional dos protocolos. Durante o handshake de SSL ou TLS, o cliente SSL ou TLS sempre obtém e valida um certificado digital do servidor. Com a implementação do WebSphere MQ , o servidor SSL ou TLS sempre solicita um certificado do cliente.

Nos sistemas UNIX, Linux, and Windows , o cliente SSL ou TLS envia um certificado somente se ele tiver um rotulado no formato correto do WebSphere MQ , que é `ibmwebspheremq` seguido por seu ID do usuário de logon alterado para minúsculas, por exemplo, `ibmwebspheremqmyuserid`.

WebSphere MQ usa o prefixo `ibmwebspheremq` em um rótulo para evitar confusão com certificados para outros produtos. Assegure-se de especificar o rótulo inteiro do certificado em minúsculas.

O servidor SSL ou TLS sempre valida o certificado de cliente se um for enviado. Se o cliente não enviar um certificado, a autenticação falha se a extremidade do canal que está agindo como o servidor SSL ou TLS estiver definida com o parâmetro SSLCAUTH configurado para REQUIRED ou um conjunto de valores do parâmetro SSLPEER. Para obter informações adicionais sobre como conectar a um gerenciador de filas de forma anônima, consulte “Conectando um cliente a um gerenciador de filas de forma anônima” na página 220.

Usando certificados autoassinados para autenticação mútua de um cliente e de um gerenciador de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre um cliente e um gerenciador de filas, usando certificados SSL ou TLS autoassinados.

Sobre esta tarefa

Cenário:

- Você tem um cliente, C1, e um gerenciador de filas, QM1, que precisam se comunicar de forma segura. É necessário executar a autenticação mútua entre C1 e QM1.
- Você decidiu testar a comunicação segura usando certificados autoassinados.

O DCM no IBM i não suporta certificados autoassinados, portanto, esta tarefa não é aplicável em sistemas IBM i.

A configuração resultante é semelhante à seguinte:

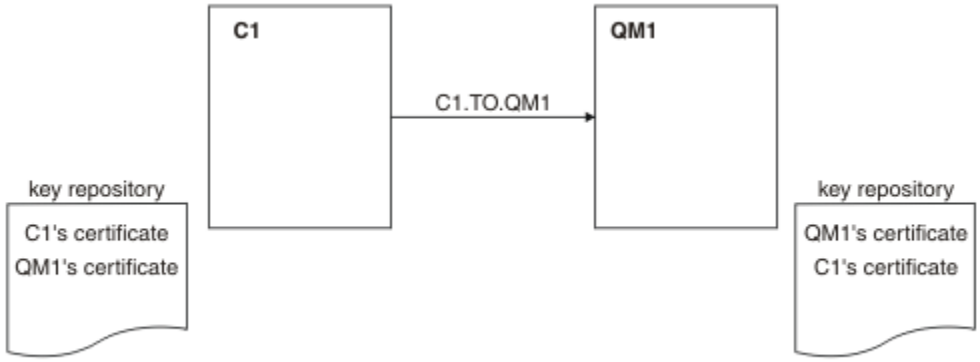


Figura 23. Configuração resultante desta tarefa

Na Figura 17 na página 217, o repositório de chaves para QM1 contém o certificado para QM1 e o certificado público de C1. O repositório de chaves para C1 contém o certificado para C1 e o certificado público de QM1.

Procedimento

1. Prepare o repositório de chaves no cliente e no gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Crie certificados autoassinados para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
3. Extraia uma cópia de cada certificado:
   - Em UNIX, Linuxe Windows sistemas
4. Transfira a parte pública do certificado C1 para o sistema QM1 e vice-versa, usando um utilitário como FTP.

5. Inclua o certificado de parceiro no repositório de chaves para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
6. Emita o comando REFRESH SECURITY TYPE(SSL) no gerenciador de filas.
7. Defina um canal de conexão do cliente de uma das seguintes maneiras:
   - Usando a chamada MQCONNX com a estrutura MQSCO no C1, conforme descrito em Criando um canal de conexão do cliente no cliente MQI do WebSphere MQ.
   - Usando uma tabela de definição de canal de cliente, conforme descrito em Criando definições de conexão do servidor e de conexão do cliente no servidor ..
8. No QM1, defina um canal de conexão do servidor, emitindo um comando como o seguinte exemplo:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
```

   O canal deve ter o mesmo nome que o canal de conexão do cliente definido na etapa 6 e usar o mesmo CipherSpec.

Resultados

Repositórios de chaves e canais são criados conforme ilustrado em Figura 17 na página 217

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                       CURRENT
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                           SUBSTATE(RECEIVE)
```

É opcional configurar o atributo de filtro SSLPEER das definições de canal. Se a definição de canal SSLPEER estiver configurada, seu valor deverá corresponder ao DN do sujeito no certificado parceiro que foi criado na Etapa 2. Após uma conexão bem-sucedida, o campo SSLPEER na saída DISPLAY CHSTATUS mostra o DN do assunto do certificado de cliente remoto

Usando certificados assinados por CA para autenticação mútua de um cliente e de um gerenciador de filas

Siga estas instruções de amostra para implementar a autenticação mútua entre um cliente e um gerenciador de filas, usando certificados SSL ou TLS assinados por CA.

Sobre esta tarefa

Cenário:

- Você tem um cliente, C1, e um gerenciador de filas, QM1, que precisam se comunicar de forma segura. É necessário executar a autenticação mútua entre C1 e QM1.
- No futuro, você está planejando usar essa rede em um ambiente de produção e, portanto, decidiu usar os certificados assinados por CA desde o início.

A configuração resultante é semelhante à seguinte:

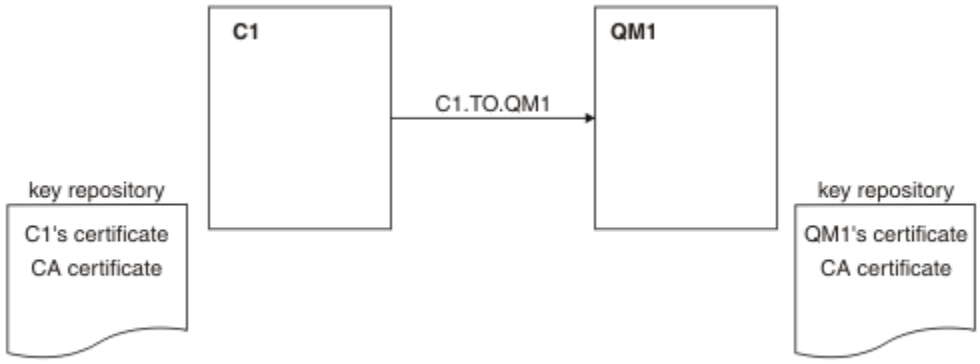


Figura 24. Configuração resultante desta tarefa

Na Figura 18 na página 219, o repositório de chaves para C1 contém certificado para C1 e o certificado de CA. O repositório de chaves para QM1 contém o certificado para QM1 e o certificado de CA. Neste exemplo, o certificado de C1 e o certificado de QM1 foram emitidos pela mesma CA. Se o certificado de C1 e o certificado de QM1 foram emitidos por CAs diferentes, os repositórios de chaves para C1 e QM1 deverão conter certificados de CA.

Procedimento

1. Prepare o repositório de chaves no cliente e no gerenciador de filas, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas
2. Solicite um certificado assinado por CA para o cliente e o gerenciador de filas.

   É possível usar CAs diferentes para o cliente e o gerenciador de filas.
   - Em UNIX, Linuxe Windows sistemas
3. Inclua o certificado de autoridade de certificação no repositório de chaves para o cliente e gerenciador de filas.

   Se o cliente e o gerenciador de filas estiverem usando diferentes Autoridades de certificação, o certificado de CA para cada Autoridade de certificação deverá ser incluído nos dois repositórios de chaves.
   - Em UNIX, Linuxe Windows sistemas
4. Inclua o certificado assinado por CA no repositório de chaves para o cliente e o gerenciador de filas:
   - Em UNIX, Linuxe Windows sistemas
5. Defina um canal de conexão do cliente de uma das seguintes maneiras:
   - Usando a chamada MQCONNX com a estrutura MQSCO no C1, conforme descrito em Criando um canal de conexão do cliente no cliente MQI do WebSphere MQ.
   - Usando uma tabela de definição de canal de cliente, conforme descrito em Criando definições de conexão do servidor e de conexão do cliente no servidor ..
6. No QM1, defina um canal de conexão do servidor emitindo um comando como o seguinte exemplo:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   O canal deve ter o mesmo nome que o canal de conexão do cliente definido na etapa 6 e usar o mesmo CipherSpec.

Resultados

Os repositórios de chaves e os canais são criados, conforme ilustrado na Figura 18 na página 219.

Como proceder a seguir

Verifique se a tarefa foi concluída com sucesso usando comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir.

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

A saída resultante é semelhante ao seguinte exemplo:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

O campo SSLPEER na saída DISPLAY CHSTATUS mostra o DN do sujeito do certificado cliente remoto que foi criado na Etapa 2. O nome do emissor corresponde ao DN do sujeito do certificado CA que assinou o certificado pessoal incluído na Etapa 4.

Conectando um cliente a um gerenciador de filas de forma anônima

Siga estas instruções de amostra para modificar um sistema com autenticação mútua para permitir que um gerenciador de filas se conecte a outro de forma anônima.

Sobre esta tarefa

Cenário:

- O gerenciador de filas e o cliente (QM1 e C1) foram configurados conforme descrito em "Usando certificados assinados por CA para autenticação mútua de um cliente e de um gerenciador de filas" na página 218.
- Você deseja alterar C1 para que ele se conecte de forma anônima a QM1.

A configuração resultante é semelhante à seguinte:

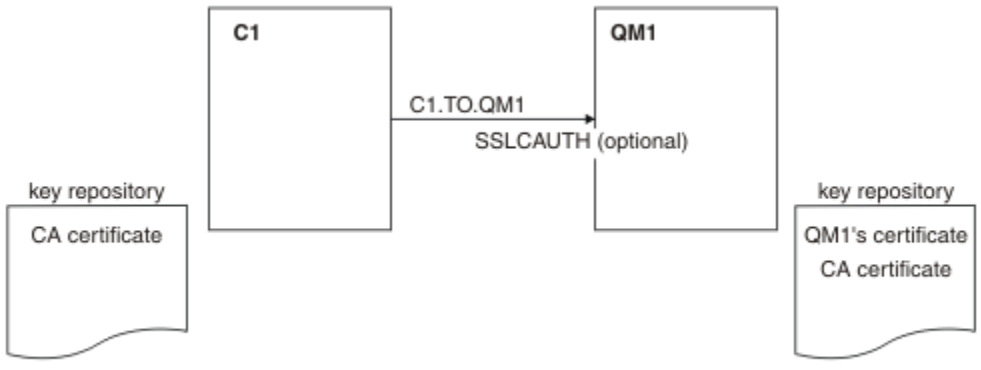


Figura 25. Cliente e gerenciador de filas permitindo conexão anônima

Procedimento

1. Remova o certificado pessoal do repositório de chaves para C1, de acordo com o sistema operacional:
   - Em UNIX, Linuxe Windows sistemas O certificado é rotulado como a seguir:
     - `ibmwebspheremq` seguido por seu ID do usuário de logon dobrado para minúsculas, por exemplo `ibmwebspheremqmyuserid`.
2. Reinicie o aplicativo cliente ou faça o aplicativo cliente fechar e reabrir todas as conexões SSL ou TLS.
3. Permita conexões anônimas no gerenciador de filas, emitindo o seguinte comando:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Resultados

Os repositórios de chaves e canais são mudados conforme ilustrado na Figura 19 na página 220

Como proceder a seguir

Na extremidade do canal do servidor, a presença do valor de parâmetro do nome do peer na exibição de status do canal indica que um certificado de cliente foi emitido.

Verifique se a tarefa foi concluída com sucesso, emitindo alguns comandos DISPLAY. Se a tarefa tiver sido bem-sucedida, a saída resultante será semelhante à mostrada no exemplo a seguir:

No gerenciador de filas QM1, insira o seguinte comando:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

A saída resultante será semelhante ao exemplo a seguir:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI( )                             SSLPEER( )
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

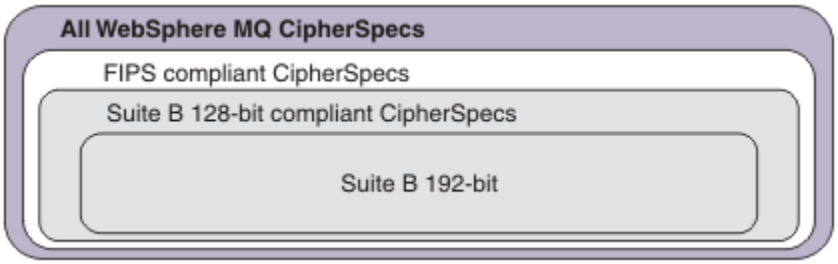
Os campos SSLCERTI e SSLPEER estão vazios, mostrando que o C1 não enviou um certificado.

Especificando CipherSpecs

Especifique um CipherSpec usando o parâmetro **SSLCIPH** no comando MQSC **DEFINE CHANNEL** ou no comando MQSC **ALTER CHANNEL** .

Alguns dos CipherSpecs que podem ser usados com IBM WebSphere MQ são compatíveis com FIPS. Outros, como NULL_MD5, não. Da mesma forma, alguns dos CipherSpecs compatíveis com FIPS também são compatíveis com o Conjunto B, embora outros não sejam. Todos os CipherSpecs compatíveis com o Conjunto B também são compatíveis com FIPS. Todos os CipherSpecs compatíveis com o Suite B caem em dois grupos: 128 bit (por exemplo, ECDHE_ECDSA_AES_128_GCM_SHA256) e 192 bit (por exemplo, ECDHE_ECDSA_AES_256_GCM_SHA384),

O diagrama a seguir ilustra o relacionamento entre estes subconjuntos:



As especificações de cifra que podem ser usadas com o suporte SSL e TLS do IBM WebSphere MQ são lisadas na tabela a seguir: Ao exigir um certificado pessoal, você especifica um tamanho de chave para o par de chaves público e particular. O tamanho de chave que é usado durante o handshake SSL é o tamanho armazenado no certificado, a menos que ele seja determinado pelo CipherSpec, conforme indicado na tabela.

Nome do CipherSpec	**Protocolo utilizado**	**Algoritmo MAC**	**Algoritmo de criptografia**	**Bits de Criptografia**	**FIPS[1]**	**Conjunto B de 128 bits**	**Conjunto B de 192 bits**
NULL_MD5 [a]	SSL 3.0	MD5	Nenhum	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Nenhum	0	No	No	No
RC4_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC4	40	No	No	No
RC4_MD5_US [a]	SSL 3.0	MD5	RC4	128	No	No	No
RC4_SHA_US [a]	SSL 3.0	SHA-1	RC4	128	No	No	No
RC2_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC2	40	No	No	No
DES_SHA_EXPORT [2] [a]	SSL 3.0	SHA-1	DES	56	No	No	No
RC4_56_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	RC4	56	No	No	No
DES_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	DES	56	No	No	No
TLS_RSA_WITH_AES_128_CBC_SHA [a]	TLS 1.0	SHA-1	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sim	No	No
TLS_RSA_WITH_DES_CBC_SHA [a]	TLS 1.0	SHA-1	DES	56	Não [5]	No	No
FIPS_WITH_DES_CBC_SHA [b]	SSL 3.0	SHA-1	DES	56	Não [6]	No	No
TLS_RSA_WITH_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sim	No	No
ECDHE_ECDSA_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No
ECDHE_RSA_RC4_128_SHA256 [b]	TLS 1.2	SHA_1	RC4	128	No	No	No

Nome do CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de criptografia	Bits de Criptografia	FIPS[1]	Conjunto B de 128 bits	Conjunto B de 192 bits
ECDHE_ECDSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sim	No	No
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sim	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sim	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	Sim	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	Sim
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sim	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sim	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Nenhum	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nenhum	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Nenhum	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Nenhum	Nenhum	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Nome do CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de criptografia	Bits de Criptografia	FIPS[1]	Conjunto B de 128 bits	Conjunto B de 192 bits

Notas:

1. Especifica se o CipherSpec é certificado por FIPS em uma plataforma certificada por FIPS. Consulte Federal Information Processing Standards (FIPS) para obter uma explicação do FIPS.
2. O tamanho de chave de handshake máximo é 512 bits. Caso nenhum dos certificados trocados durante o protocolo de reconhecimento do SSL tenha um tamanho de chave superior a 512 bits, uma chave de 512 bits temporária será gerada para uso durante o protocolo de reconhecimento.
3. O tamanho de chave de handshake é 1024 bits.
4. Esse CipherSpec não pode ser usado para proteger uma conexão do WebSphere MQ Explorer com um gerenciador de filas, a menos que os arquivos de políticas irrestritos apropriados sejam aplicados ao JRE usado pelo Explorer.
5. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007.
6. Este CipherSpec era certificado FIPS 140-2 antes de 19 de Maio de 2007. O nome do `FIPS_WITH_DES_CBC_SHA` é histórico e reflete o fato de que esse CipherSpec era anteriormente (mas não é mais) compatível com FIPS. Esse CipherSpec foi descontinuado e seu uso não é recomendado.
7. O CipherSpec pode ser usado para transferir até 32 GB de dados antes de a conexão ser finalizada com o erro AMQ9288. Para evitar esse erro, evite o uso de DES triplo ou ative a reconfiguração de chave secreta ao usar esse CipherSpec.

Suporte da Plataforma:

- a Disponível em todas as plataformas suportadas
- b Disponível apenas em plataformas UNIX, Linux, and Windows

Conceitos relacionados

“Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ” na página 35
Este tópico fornece informações sobre como escolher os certificados digitais e do CipherSpecs para sua política de segurança, delineando o relacionamento entre os certificados digitais e de CipherSpecs no IBM WebSphere MQ.

Referências relacionadas

DEFINE CHANNEL
ALTER CHANNEL

Obtendo informações sobre CipherSpecs usando o IBM WebSphere MQ Explorer

É possível usar o IBM WebSphere MQ Explorer para exibir descrições de CipherSpecs.

Utilize o seguinte procedimento para obter informações sobre o CipherSpecs no “Especificando CipherSpecs” na página 221:

1. Abra o **IBM WebSphere MQ Explorer** e expanda a pasta **Gerenciadores de filas**.
2. Certifique-se de que iniciou o gerenciador de filas.
3. Selecione o Gerenciador de Filas com o qual deseja trabalhar e clique em **Canais**.
4. Clique com o botão direito no canal que deseja trabalhar e selecione **Propriedades**.
5. Selecione a página de propriedades **SSL**.
6. Selecione da lista o CipherSpec com o qual quer trabalhar. Uma descrição é exibida na janela abaixo da lista.

Alternativas para a Especificação do CipherSpecs

Para as plataformas em que o sistema operacional fornece suporte ao SSL, é possível que o sistema suporte novos CipherSpecs. Você pode especificar um novo CipherSpec com o parâmetro SSLCIPH, mas o valor que você fornecer dependerá da sua plataforma.

Nota: Esta seção não se aplica a sistemas UNIX, Linux ou Windows , porque os CipherSpecs são fornecidos com o produto WebSphere MQ , portanto, os novos CipherSpecs não se tornam disponíveis após a remessa

Para aquelas plataformas em que o sistema operacional fornece suporte ao SSL, é possível que o seu sistema suporte os novos CipherSpecs que não são incluídos no “Especificando CipherSpecs” na página 221. Você pode especificar um novo CipherSpec com o parâmetro SSLCIPH, mas o valor que você fornecer dependerá da sua plataforma. Em todos os casos, a especificação *deve* corresponder a um SSL CipherSpec que seja válido e suportado pela versão do SSL com o qual seu sistema está sendo executado.

IBM i

Uma cadeia de dois caracteres que representa um valor hexadecimal.

Para obter mais informações sobre os valores permitidos, consulte a documentação apropriada do produto (procure *cipher_spec* na documentação do produtoIBM i).

Você pode utilizar o comando CHGMQMCHL ou CRTMQMCHL para especificar o valor, por exemplo:

```
CRTMQMCHL CHLNAME('channel name') SSLCIPH('hexadecimal value')
```

Também é possível usar o comando ALTER QMGR MQSC para configurar o parâmetro SSLCIPH.

z/OS

Uma cadeia de dois caracteres que representa um valor hexadecimal. Os códigos hexadecimais correspondem aos valores definidos no protocolo SSL.

Para obter mais informações, consulte a descrição de gsk_environment_open () no capítulo de referência da API do *z/OS Cryptographic Services System SSL Programming*, SC24-5901, em que há uma lista de todas as especificações de código SSL V3.0 e TLS V1.0 suportadas no formato de códigos hexadecimais de 2 dígitos.

Considerações para clusters do WebSphere MQ

Com clusters do WebSphere MQ , é mais seguro usar os nomes CipherSpec em “Especificando CipherSpecs” na página 221. Se você usar uma especificação alternativa, tenha em mente que a especificação pode não ser válida em outras plataformas. Para obter informações adicionais, consulte “SSL e clusters” na página 253.

Especificando um CipherSpec para um cliente MQI do IBM WebSphere MQ

Você tem três opções para especificar um CipherSpec para um cliente MQI do IBM WebSphere MQ

Estas opções são as seguintes:

- Utilizando uma tabela de definição de canais
- Usando o campo SSLCipherSpec na estrutura MQCD, no MQCD_VERSION_7 ou mais recente, em uma chamada MQCONNX.
- Usando o Active Directory (em sistemas Windows com suporte do Active Directory)

Especificando um CipherSuite com classes IBM WebSphere MQ para Java e classes IBM WebSphere MQ para JMS

As classes IBM WebSphere MQ para Java e IBM WebSphere MQ para JMS especificam CipherSuites de forma diferente de outras plataformas.

Para obter informações sobre como especificar um CipherSuite com IBM WebSphere MQ classes para Java, consulte Suporte SSL (Secure Sockets Layer).

Para obter informações sobre como especificar um CipherSuite com IBM WebSphere MQ classes para JMS, consulte Usando Secure Sockets Layer (SSL) com WebSphere MQ classes para JMS.

Auditing

É possível procurar por intrusões de segurança ou tentativas de intrusão usando mensagens do evento. Também é possível verificar a segurança do seu sistema usando o IBM WebSphere MQ Explorer.

Para detectar tentativas de executar ações não autorizadas como conectar-se a um gerenciador de filas ou colocar uma mensagem em uma fila, examine as mensagens de eventos produzidas por seus gerenciadores de filas, sobretudo mensagens de eventos de autoridade. Para obter mais informações sobre mensagens do gerenciador de filas, consulte Eventos do gerenciador de filase, para obter mais informações sobre o monitoramento de eventos em geral, consulte Monitoramento de eventos .

Mantendo Clusters Seguros

Autorize ou evite que os gerenciadores de filas juntem os clusters ou coloquem as mensagens nas filas de cluster. Force um gerenciador de filas para deixar um cluster. Leve em conta algumas considerações adicionais ao configurar o SSL para clusters.

Parando o envio de mensagens por gerenciadores de filas desautorizados

Evite que os gerenciadores de filas desautorizados enviem mensagens para seu gerenciador de filas usando uma saída de segurança do canal.

Antes de começar

O armazenamento em cluster não tem efeito na maneira como as saídas de segurança funcionam. Você pode restringir o acesso a um gerenciador de filas da mesma maneira que faria em um ambiente de enfileiramento distribuído.

Sobre esta tarefa

Evite que os gerentes de filas selecionados enviem mensagens ao seu gerenciador de filas:

Procedimento

1. Defina um programa de saída de segurança do canal na definição de canal CLUSRCVR.
2. Grave um programa que autentica gerenciadores de filas que estão tentando enviar mensagens em seu canal do receptor de clusters e nega-lhes o acesso se não estiverem autorizados.

Como proceder a seguir

Os programas de saída de segurança do canal são chamados na inicialização e na rescisão de MCA.

Parando a Colocação de Mensagens de Gerenciadores de Filas Desautorizados em suas Filas

Use o canal para colocar o atributo de autoridade no canal do receptor de clusters para parar os gerenciadores de filas não autorizados de colocar mensagens nas suas filas. Autorize um gerenciador de filas remotas verificando o ID do usuário na mensagem usando RACF no z/OSou o OAM em outras plataformas.

Sobre esta tarefa

Use os recursos de segurança de uma plataforma e o mecanismo de controle de acesso no WebSphere MQ para controlar o acesso às filas

Procedimento

1. Para evitar que determinados gerenciadores de filas coloquem mensagens em uma fila, use os recursos de segurança disponíveis em sua plataforma.

   Por exemplo:

   - RACF ou outros gerenciadores de segurança externos no WebSphere MQ para z/OS
   - O gerenciador de autoridade de objeto (OAM) em outras plataformas.
2. Use a autoridade put, PUTAUT, o atributo CLUSRCVR na definição de canal.

   O atributo PUTAUT permite que você especifique quais os identificadores de usuário devem ser usados para estabelecer a autoridade para colocar uma mensagem em uma fila.

   As opções no atributo PUTAUT são:

   DEF
   Use o ID do usuário padrão. No z/OS, a verificação pode envolver o uso do ID do usuário recebido da rede e derivado de MCAUSER.

   CTX
   Use o ID do usuário nas informações de contexto associadas à mensagem. No z/OS , a verificação pode envolver o uso do ID do usuário recebido da rede ou derivado de MCAUSERou ambos. Use esta opção se o link for confiável e autenticado.

   ONLYMCA (somentez/OS)
   Igual a DEF, mas qualquer ID do usuário recebido da rede não é usado. Use esta opção se o link não for confiável. Você deseja permitir somente um conjunto específico de ações nele, que são definidos para o MCAUSER.

   ALTMCA (somentez/OS)
   Como para CTX, mas qualquer ID do usuário recebido da rede não é usado.

Autorizando a Colocação de Mensagens em Filas de Cluster Remotas

Em sua plataforma, autorize o acesso para conectar-se ao gerenciador de fila e para colocar na fila nesse gerenciador de fila

Sobre esta tarefa

O comportamento padrão é executar controle de acesso com relação a SYSTEM.CLUSTER.TRANSMIT.QUEUE. Observe que esse comportamento se aplica mesmo se você estiver usando diversas filas de transmissão.

O comportamento específico descritos neste tópico se aplica somente quando você tiver configurado o atributo **ClusterQueueAccessControl** no arquivo qm.ini para ser *RQMName*, conforme descrito no tópico Sub-rotina de segurança e reiniciou o gerenciador de filas.

Procedimento

- Para sistemas UNIX, Linux e Windows , emita os comandos a seguir:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

  O usuário pode colocar mensagens apenas na fila de clusters especificada, e em nenhuma outra fila de clusters.

  Os nomes das variáveis possuem os seguintes significados:

QMgrName
O nome do gerenciador de filas.

GroupName
O nome do grupo que receberá acesso.

QueueName
Nome da fila ou perfil genérico para o qual mudar as autorizações.

Como proceder a seguir

Se você especificar uma fila de resposta quando colocar uma mensagem em uma fila de clusters, o aplicativo consumidor deverá ter autoridade para enviar a resposta. Configure essa autoridade seguindo as instruções em "Concedendo autoridade para colocar mensagens em uma fila do cluster remoto" na página 196.

Informações relacionadas

Sub-rotina de segurança no qm.ini

Impedindo que Gerenciadores de Filas se Juntem a um Cluster

Se um gerenciador de filas nocivo se unir a um cluster é difícil evitar o recebimento de mensagens que você não deseja que ele receba.

Procedimento

Se você desejar assegurar que somente determinados gerenciadores de filas autorizados se juntar a um cluster, você tem a opção de três técnicas:

- Usar registros de autenticação de canal é possível bloquear a conexão do canal do cluster com base em: o endereço IP remoto, o nome do gerenciador de filas remotas ou o Nome distinto SSL/TLS fornecido pelo sistema remoto.
- Gravar um programa de saída para evitar que gerenciadores de filas não autorizados gravem em `SYSTEM.CLUSTER.COMMAND.QUEUE`. Não restrinja o acesso a `SYSTEM.CLUSTER.COMMAND.QUEUE` de forma que nenhum gerenciador de filas possa gravar nele ou você impediria que qualquer gerenciador de filas se junte ao cluster.
- Um programa de saída de segurança na definição de canal `CLUSRCVR`.

Saídas de segurança nos canais de cluster

Considerações extra ao usar saídas de segurança em canais de cluster.

Sobre esta tarefa

Quando um canal do emissor de clusters é iniciado pela primeira vez, ele usa atributos definidos manualmente por um administrador do sistema. Quando o canal é interrompido e reiniciado, ele seleciona os atributos da definição do canal do receptor de clusters correspondente. A definição de canal do emissor de clusters original é sobrescrita com os novos atributos, incluindo o atributo `SecurityExit`.

Procedimento

1. Deve-se definir uma saída de segurança em ambas as extremidades do emissor de cluster e o receptor de cluster de um canal.

   A conexão inicial deve ser feita com um handshake de saída de segurança, mesmo que o nome de saída de segurança seja enviado a partir da definição do receptor de cluster.
2. Valide o `PartnerName` na estrutura `MQCXP` na saída de segurança.

   A saída deve permitir que o canal inicie somente se o gerenciador de filas do parceiro estiver autorizado
3. Projete a saída de segurança na definição do receptor de cluster para ser iniciada pelo destinatário.

4. Se você projetá-la como iniciada pelo emissor, um gerenciador de filas não autorizado sem uma saída de segurança poderá unir-se ao cluster porque nenhuma verificação de segurança será executada.

   Não até o canal ser interrompido e reiniciado pode o nome SCYEXIT ser enviado a partir da definição do receptor de cluster e verificações de segurança integral feita.
5. Para visualizar a definição de canal do emissor de clusters que está atualmente em uso, use o comando:

   ```
   DISPLAY CLUSQMGR(queue manager) ALL
   ```

   O comando exibe os atributos que foram enviados a partir da definição do receptor de clusters.
6. Para visualizar a definição original, use o comando:

   ```
   DISPLAY CHANNEL(channel name) ALL
   ```
7. Você pode precisar definir uma saída de autodefinição de canal CHADEXIT, no gerenciador de filas do emissor de cluster, se os gerenciadores de filas estiverem em plataformas diferentes.

   Use a saída de definição automática de canal para configurar o atributo SecurityExit para um formato apropriado para a plataforma de destino.
8. Implementar e configurar a saída de segurança.

   Windows, UNIX and Linux sistemas

   - A biblioteca de vínculo dinâmico de saída de segurança deve estar no caminho especificado no atributo SCYEXIT da definição de canal.
   - A biblioteca de vínculo dinâmico de saída de autodefinição de canal deve estar no caminho especificado no atributo CHADEXIT da definição do gerenciador de filas.

Forçando Gerenciadores de Filas Indesejáveis a Sair de um Cluster

Forçar um gerenciador de filas indesejado para deixar um cluster, emitindo o comando RESET CLUSTER em um gerenciador de filas de repositório completo.

Sobre esta tarefa

É possível forçar um gerenciador de filas indesejado para deixar um cluster. Se, por exemplo, um gerenciador de filas for excluído mas seus canais do receptor de clusters ainda estiverem definidos no cluster. Você pode desejar organizar.

Somente gerenciadores de fila de repositório completo têm autorização para ejetar um gerenciador de filas de um cluster.

Siga este procedimento para ejetar o gerenciador de filas OSLO do cluster NORWAY:

Procedimento

1. Em um gerenciador de filas de repositório completo, emita o comando:

   ```
   RESET CLUSTER(NORWAY) QMNAME(OSLO) ACTION(FORCEREMOVE)
   ```
2. Como alternativa use o QMID em vez de QMNAME no comando:

   ```
   RESET CLUSTER(NORWAY) QMID(qmid) ACTION(FORCEREMOVE)
   ```

Resultados

O gerenciador de filas que é removido pela força não é alterado: suas definições de cluster local mostram que ele está no cluster.. As definições em todos os outros gerenciadores de filas não mostram isso no cluster.

Impedindo que gerenciadores de filas recebam mensagens

É possível evitar que um gerenciador de filas do cluster receba mensagens que ele não está autorizado a receber, usando programas de saída.

Sobre esta tarefa

É difícil parar a definição de uma fila por um gerenciador de filas que é membro de um cluster. Há um risco de que um gerenciador de filas nocivo se una a um cluster e defina sua própria instância de uma das filas no cluster. Agora ele pode receber mensagens que ele não está autorizado a receber. Para impedir que um gerenciador de filas receba mensagens, use uma das seguintes opções determinado no procedimento.

Procedimento

- Um programa de saída do canal em cada canal do emissor de clusters. O programa de saída usa o nome de conexão para determinar a adequação do gerenciador de filas de destino para que as mensagens sejam enviadas.
- Um programa de saída de carga de do cluster que usa os registros de destino para determinar a adequação da fila de destino e do gerenciador de filas para ser enviadas as mensagens.

SSL e clusters

Ao configurar o SSL para clusters, esteja ciente que uma definição de canal CLUSRCVR é propagada para outros gerenciadores de filas como um canal CLUSSDR definido automaticamente. Se um canal CLUSRCVR usa SSL, deve-se configurar o SSL em todos os gerenciadores de filas que se comunicam usando o canal.

Para obter mais informações sobre SSL, consulte o suporte do WebSphere MQ para SSL e TLS O conselho lá geralmente é aplicável aos canais do cluster, mas talvez você deseje fornecer alguma consideração especial ao seguinte:

Em um cluster do IBM WebSphere MQ uma determinada definição de canal CLUSRCVR é frequentemente propagada a muitos outros gerenciadores de filas nos quais é transformada em um CLUSSDR definido automaticamente. Subsequentemente o CLUSSDR definido automaticamente é usado para iniciar um canal para o CLUSRCVR. Se o CLUSRCVR for configurado para conectividade de SSL, as considerações a seguir se aplicam:

- Todos os gerenciadores de filas que desejam se comunicar com este CLUSRCVR devem ter acesso ao suporte a SSL. Esta provisão de SSL deve suportar CipherSpec para o canal.
- Os diferentes gerenciadores de filas para os quais os canais do emissor de clusters definidos automaticamente foram propagados terão, cada, um nome distinto diferente associado. Se a verificação de peer de nome distinto deve ser usada no CLUSRCVR ela deve ser configurada para que todos os nomes distintos que serão recebidos sejam correspondidos com sucesso.

  Por exemplo, vamos assumir que todos os gerenciadores de filas que hospedarão os canais do emissor de clusters que se conectarão a um determinado CLUSRCVR possuem certificados associados. Também vamos assumir que os nomes distintos em todos estes certificados definam o país como UK, a organização como IBM, a unidade da organização como IBM WebSphere MQ Development e que todos possuam nomes comuns no formato DEVT.QMnnn, em que nnn é numérico.

  Nesse caso, um valor SSLPEER de C=UK, O=IBM, OU=WebSphere MQ Development, CN=DEVT.QM* no CLUSRCVR permitirá que todos os canais do emissor de clusters necessários se conectem com êxito, mas evitará que os canais do emissor de clusters indesejados se conectem.
- Se cadeias de CipherSpec customizadas forem usadas, esteja ciente de que os formatos de cadeia customizados não são permitidos em todas as plataformas. Um exemplo disto é que a sequência CipherSpec RC4_SHA_US tem um valor de 05 no IBM i mas não é uma especificação válida nos sistemas UNIX, Linux ou Windows. Portanto, se os parâmetros SSLCIPH customizados são usados em uma CLUSRCVR, todos os canais do emissor de clusters definidos automaticamente resultantes deverão residir nas plataformas nas quais o suporte a SSL subjacente implementa este CipherSpec

e nas quais ele pode ser especificado com o valor customizado. Se não for possível selecionar um valor para o parâmetro SSLCIPH que será entendido em todo o seu cluster, será preciso uma saída de autodefinição de canal para mudá-lo para algo que as plataformas que estão sendo usadas entendam. Use as sequências textuais CipherSpec nas quais é possível (por exemplo RC4_MD5_US).

Um parâmetro SSLCRLNL se aplica a um gerenciador de filas individual e não é propagado a outros gerenciadores de filas em um cluster.

Fazendo upgrade de gerenciadores de filas em cluster e canais para SSL

Faça upgrade dos canais do cluster um por vez, mudando todos os canais CLUSRCVR antes dos canais CLUSSDR.

Antes de começar

Considere as considerações a seguir, como elas podem afetar sua escolha de CipherSpec para um cluster:

- Alguns CipherSpecs não estão disponíveis em todas as plataformas. Tome cuidado ao escolher um CipherSpec que é suportado por todos os gerenciadores de filas no cluster.
- Alguns CipherSpecs podem ser novos na liberação atual do WebSphere MQ e não são suportados em liberações mais antigas Um cluster contendo gerenciadores de filas em execução em diferentes liberações do MQ somente poderá usar os CipherSpecs suportados por cada liberação.

  Para usar um novo CipherSpec dentro de um cluster, primeiro deve-se migrar todos os gerenciadores de filas do cluster para a liberação atual.
- Alguns CipherSpecs requerem um tipo específico de certificado digital para ser usado, especialmente aquelas que usam Elliptic Curve Cryptography.

Atualize todos os gerenciadores de filas no cluster para WebSphere MQ V6 ou superior, se eles ainda não estiverem nesses níveis. Distribua os certificados e chaves para que o SSL funcione a partir de cada um deles.

Sobre esta tarefa

Altere um CLUSRCVR por vez e permita que as mudanças fluam pelo cluster antes de mudar o próximo. Certifique-se de não alterar o caminho reverso até que as alterações para o canal atual tenham sido distribuídas por todo o cluster.

Procedimento

1. Alterne os canais CLUSRCVR para SSL em qualquer ordem.

   As mudanças fluem na direção oposta sobre canais que não são alterados para SSL.
2. Alterne todos os canais CLUSSDR manuais para SSL.

   Isto não possui qualquer efeito na operação do cluster, a menos que você use o comando REFRESH CLUSTER com a opção REPOS(YES).

   Nota: Para grandes clusters, o uso do comando **REFRESH CLUSTER** pode ser disruptivo para o cluster enquanto ele está em andamento e novamente em intervalos de 27 dias, quando os objetos de cluster enviam automaticamente atualizações de status para todos os gerenciadores de filas de seu interesse. Consulte Atualizando em um grande cluster pode afetar o desempenho e disponibilidade do cluster.

Conceitos relacionados

“Especificando CipherSpecs” na página 221
Especifique um CipherSpec usando o parâmetro **SSLCIPH** no comando MQSC **DEFINE CHANNEL** ou no comando MQSC **ALTER CHANNEL** .

“Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ” na página 35
Este tópico fornece informações sobre como escolher os certificados digitais e do CipherSpecs para sua política de segurança, delineando o relacionamento entre os certificados digitais e de CipherSpecs no IBM WebSphere MQ.

Informações relacionadas

Armazenamento em Cluster: Usando Melhores Práticas de REFRESH CLUSTER

Desativando SSL ou TLS em gerenciadores de filas e canais em cluster

Para desativar SSL ou TLS, configure o parâmetro SSLCIPH como `' '`. Desative TLS nos canais de cluster individualmente, mudando todos os canais do receptor de clusters antes dos canais do emissor de clusters.

Sobre esta tarefa

Mude um canal do receptor de clusters por vez e permita que as mudanças fluam através do antes de mudar o próximo.

Importante: Assegure-se de não mudar o caminho reverso até que as mudanças para o canal atual tenham sido distribuídas por todo o cluster.

Procedimento

1. Configure o valor do parâmetro SSLCIPH como `' '`, uma cadeia vazia entre aspas .

   É possível desativar SSL ou TLS nos canais receptores do cluster em qualquer ordem que você desejar

   Observe que as mudanças fluem na direção oposta em canais nos quais você deixa SSL ou TLS ativo.
2. Verifique se o novo valor é refletido em todos os outros Gerenciadores de Filas usando o comando **DISPLAY CLUSQMGR(*)** ALL.
3. Desative SSL ou TLS em todos os canais do emissor de cluster manuais.

   Isto não possui qualquer efeito na operação do cluster, a menos que você use o comando **REFRESH CLUSTER** com a opção REPOS(YES).

   Para clusters grandes, o uso do comando **REFRESH CLUSTER** pode ser disruptivo para o cluster enquanto ele está em andamento e novamente em intervalos regulares, quando os objetos de cluster enviam automaticamente atualizações de status para todos os gerenciadores de filas interessados. Consulte Atualizando em um cluster grande pode afetar o desempenho e disponibilidade do cluster para obter mais informações.
4. Pare e reinicie os canais do emissor de clusters.

Segurança de Publicação/Assinatura

Os componentes e as interações que estão envolvidos na publicação/assinatura estão descritos como uma introdução para as explicações mais detalhadas e exemplos a seguir.

Existem inúmeros componentes envolvidos na publicação e assinatura para um tópico. Alguns dos relacionamentos de segurança entre eles estão ilustrados em Figura 26 na página 256 e descritos no seguinte exemplo.

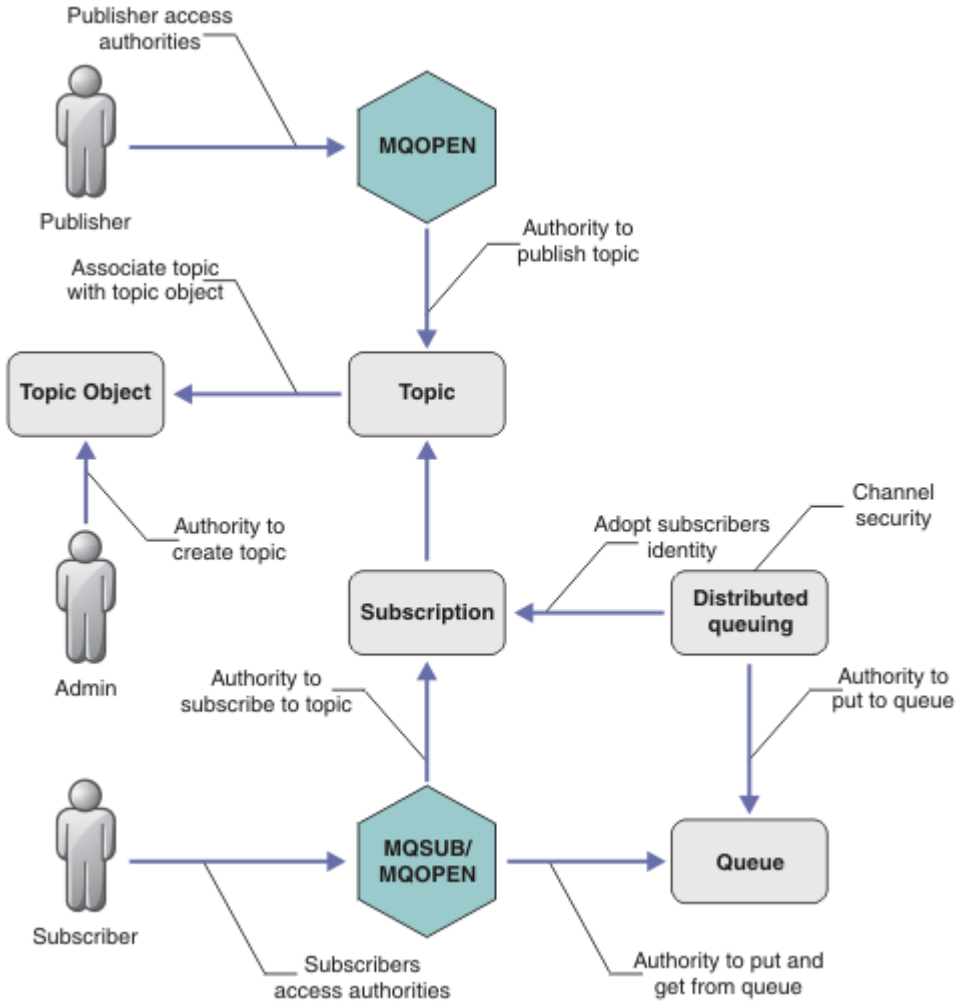


Figura 26. Relacionamentos de segurança de publicação / assinatura

Tópicos

Os tópicos são identificados por sequências de tópicos e geralmente organizados em árvores; consulte Árvores de tópicos. É necessário associar um tópico a um objeto do tópico para controlar o acesso ao tópico."Modelo de Segurança do Tópico" na página 258 explica como proteger tópicos usando objetos de tópico.

Objetos de Tópico Administrativo

É possível controlar quem tem acesso a um tópico e para qual propósito, usando o comando **setmqaut** com uma lista de objetos de tópico administrativo. Consulte os exemplos, "Conceder acesso a um usuário para assinar um tópico" na página 263 e "Conceder acesso a um usuário para publicar um tópico" na página 268.

Assinaturas

Subscreva-se em um ou mais tópicos, criando uma assinatura que fornece uma sequência de tópicos, que pode incluir curingas, para corresponder nas sequências de tópicos das publicações. Para obter detalhes adicionais, consulte:

Subscrever usando um objeto do tópico

"Subscrevendo Usando o Nome do Objeto de Tópico" na página 259

Subscrever usando um tópico

"Subscrevendo Usando a Sequência de Tópicos na Qual o Nó de Tópico não Existe" na página 260

Subscrever usando um tópico com curingas

"Subscrevendo Usando uma Sequência de Tópicos que Contém Caracteres Curinga" na página 260

Uma assinatura contém informações sobre a identidade do assinante e a identidade da fila de destino na qual as publicações devem ser colocadas. Ela também contém informações sobre como a publicação deve ser colocada na fila de destino.

Assim como definir quais assinantes possuem autoridade para se subscrever em certos tópicos, é possível restringir as assinaturas para que sejam usadas por um assinante individual. Também é possível controlar quais informações sobre o assinante são usadas pelo gerenciador de filas quando as publicações forem colocadas na fila de destino. Consulte o "Segurança de assinatura" na página 272.

Filas

A fila de destino é uma fila importante para proteger. É local para o assinante e as publicações que correspondiam à assinatura são colocadas nele. Você precisa considerar o acesso à fila de destino a partir de duas perspectivas:

1. Colocando uma publicação na fila de destino.
2. Obtendo a publicação da fila de destino.

O gerenciador de filas coloca uma publicação na fila de destino usando uma identidade fornecida pelo assinante. O assinante ou um programa que delegou a tarefa de obter as publicações, obtém as mensagens da fila. Consulte o "Autoridade para Filas de Destino" na página 261.

Não há nenhum alias de objeto do tópico, mas é possível usar uma fila de alias como o alias para um objeto do tópico. Se fizer isso, e também verificar a autoridade para usar o tópico para publicação ou assinatura, o gerenciador de filas verificará a autoridade para usar a fila.

Segurança de Publicação/Assinatura entre os Gerenciadores de Filas

Sua permissão para publicar ou subscrever em um tópico é verificada no gerenciador de filas locais usando as identidades e autorizações locais. A autorização não depende de o tópico ser definido ou não, nem de onde está definido. Consequentemente, você precisa executar a autorização de tópico em cada gerenciador de filas em um cluster quando os tópicos em cluster forem usados.

Nota: O modelo de segurança para tópicos difere do modelo de segurança para filas. É possível alcançar o mesmo resultado para as filas definindo um alias da fila localmente para cada fila em cluster.

Os gerenciadores de fila trocam assinaturas em um cluster. Na maioria das configurações de cluster do WebSphere MQ , canais são configurados com PUTAUT=DEF para colocar mensagens em filas de destino usando a autoridade do processo do canal. É possível modificar a configuração do canal para usar PUTAUT=CTX para requerer que o usuário de assinatura tenha autoridade para propagar uma assinatura para outro gerenciador de filas em um cluster.

Segurança de publicação / assinatura entre gerenciadores de filas descreve como mudar suas definições de canal para controlar quem tem permissão para propagar assinaturas para outros servidores no cluster.

Autorização

É possível aplicar autorização em objetos de tópico, assim como filas e outros objetos. Existem três operações de autorização, `pub`, `sub` e `resume` que permitem a você aplicar apenas nos tópicos. Os detalhes estão descritos em Especificando Autoridades para Diferentes Tipos de Objeto.

Chamadas de função

Nos programas de publicação e assinatura, como em programas enfileirados, as verificações de autorização são feitas quando os objetos são abertos, criados, alterados ou excluídos. As verificações não são feitas quando as chamadas `MQPUT` ou `MQGET` `MQI` são feitas para colocar e obter as publicações.

Para publicar um tópico, execute um MQOPEN no tópico, que executa as verificações de autorização. As mensagens publicadas na manipulação de tópico que usam o comando MQPUT, que não executa nenhuma autorização.

Para subscrever-se em um tópico, geralmente execute um comando MQSUB para criar ou retornar a assinatura e também para abrir a fila de destino para receber as publicações. Como alternativa, execute um MQOPEN separado para abrir a fila de destino e, em seguida, execute o MQSUB para criar ou retomar a assinatura.

Independente das chamadas que você usar, o gerenciador de filas verificar se é possível se subscrever no tópico e obter as publicações resultantes da fila de destino. Se a fila de destino não for gerenciada, as verificações de autorização também serão feitas para que o gerenciador de filas consiga colocar as publicações na fila de destino. Ela usa a identidade que adotou de uma assinatura correspondente. É assumido que o gerenciador de filas sempre consegue colocar as publicações nas filas de destino gerenciadas.

Papéis

Os usuários estão envolvidos em quatro funções na execução de aplicativos de publicação/assinatura:

1. Publisher
2. Assinante
3. Administrador do tópico
4. WebSphere MQ Administrador-membro do grupo mqm

Defina os grupos com autorizações apropriadas correspondentes às funções de administração do tópico, publicação e assinatura. Em seguida, é possível designar os diretores desses grupos autorizando-os a executar tarefas específicas de publicação e assinatura.

Além disso, você precisa estender as autorizações de operações administrativas para o administrador das filas e canais responsáveis por mover as publicações e assinaturas.

Modelo de Segurança do Tópico

Apenas os objetos de tópico definido possuem atributos de segurança associados. Para uma descrição de objetos de tópico, consulte Objetos de tópico administrativo. Os atributos de segurança especificam se um ID do usuário especificado ou grupo de segurança, tem permissão para executar uma operação de assinatura ou publicação em cada objeto do tópico.

Os atributos de segurança são associados ao nó de administração apropriado na árvore de tópicos. Quando uma verificação de autoridade é feita para um determinado ID do usuário durante uma operação de assinatura ou publicação, a autoridade concedida será baseada nos atributos de segurança do nó associado da árvore de tópicos.

Os atributos de segurança são uma lista de controle de acesso, que indica qual autoridade um determinado ID do usuário do sistema operacional ou grupo de segurança tem para o objeto do tópico.

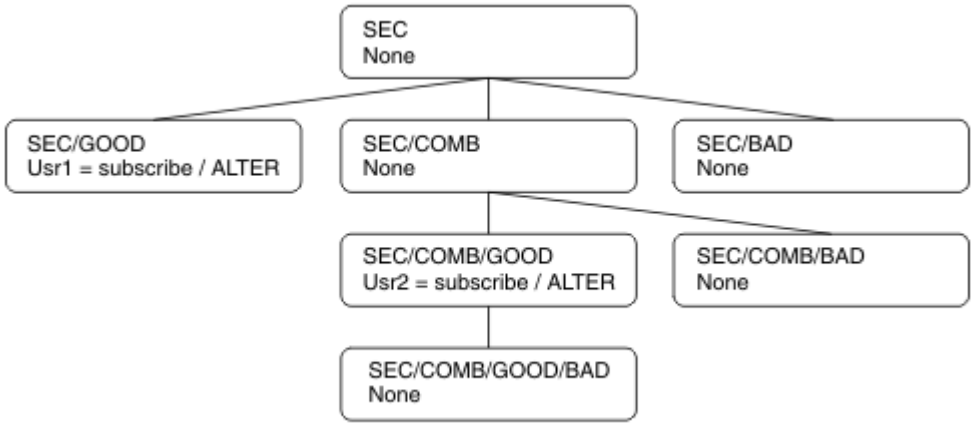
Considere o seguinte exemplo no qual os objetos do tópico foram definidos com os atributos de segurança ou autoridades mostradas:

Tabela 17. Autoridades de Objeto do Tópico de Exemplo

Nome do tópico	Cadeia do tópico	Autoridades-não z/OS	Autoridades do z/OS
SECROOT	SEC	Nenhum	Nenhum
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD
SECBAD	SEC/BAD	Nenhum	Nenhum HLQ.SUBSCRIBE.SECBAD

Tabela 17. Autoridades de Objeto do Tópico de Exemplo (continuação)			
Nome do tópico	**Cadeia do tópico**	**Autoridades-não z/OS**	**Autoridades do z/OS**
SECCOMB	SEC/COMB	Nenhum	Nenhum HLQ.SUBSCRIBE.SECCOMB
SECCOMBB	SEC/COMB/ GOOD/BAD	Nenhum	Nenhum HLQ.SUBSCRIBE.SECCOMBB
SECCOMBG	SEC/COMB/GOOD	usr2+subscribe	ALTER HLQ.SUBSCRIBE.SECCOMBG
SECCOMBN	SEC/COMB/BAD	Nenhum	Nenhum HLQ.SUBSCRIBE.SECCOMBN

A árvore de tópicos com os atributos de segurança associados em cada nó pode ser representada da seguinte maneira:



Os exemplos listados fornecem as seguintes autorizações:

- No nó-raiz da árvore /SEC, nenhum usuário tem autoridade nesse nó.
- usr1 recebeu autoridade de assinatura para o objeto /SEC/GOOD
- usr2 recebeu autoridade de assinatura para o objeto /SEC/COMB/GOOD

Subscrevendo Usando o Nome do Objeto de Tópico

Ao subscrever em um objeto do tópico especificando o nome MQCHAR48, o nó correspondente na árvore de tópicos está localizado. Se os atributos de segurança associados ao nó indicarem que o usuário tem autoridade para se subscrever, então o acesso será concedido.

Se o usuário não tiver acesso concedido, o nó pai na árvore determinará se o usuário tem autoridade para se subscrever no nível de nó pai. Em caso positivo, o acesso é concedido. Em caso negativo, o pai desse nó será considerado. A recursão continua até que seja localizado um nó que conceda autoridade de assinatura para o usuário. A recursão para quando o nó-raiz é considerado sem que a autoridade tenha sido concedida. No último caso, o acesso é negado.

Em resumo, se algum nó no caminho conceder autoridade para se subscrever nesse usuário ou aplicativo, o assinante terá permissão para se subscrever nesse nó ou em qualquer local abaixo desse nó na árvore de tópicos.

O nó-raiz no exemplo é SEC.

O usuário recebe autoridade de assinatura, se a lista de controle de acesso indicar que o ID do usuário em si tem autoridade ou se um grupo de segurança do sistema operacional do qual o ID do usuário é membro tiver autoridade.

Assim, por exemplo:

- Se `usr1` tentar se subscrever, usando uma sequência de tópicos de `SEC/GOOD`, a assinatura seria alocada porque o ID do usuário tem acesso ao nó associado a esse tópico. No entanto, se `usr1` tentou se subscrever usando a sequência de tópicos `SEC/COMB/GOOD` a assinatura não seria permitida porque o ID do usuário não tem acesso ao nó associado.
- Se o `usr2` tentar se subscrever, usando uma sequência de tópicos de `SEC/COMB/GOOD`, a assinatura seria alocada porque o ID do usuário tem acesso ao nó associado ao tópico. No entanto, se `usr2` tentou se subscrever ao `SEC/GOOD` a assinatura não seria permitida porque o ID do usuário não tem acesso ao nó associado.
- Se `usr2` tentar se subscrever usando uma sequência de tópicos de `SEC/COMB/GOOD/BAD`, a assinatura seria permitida porque o ID do usuário tem acesso ao nó-pai `SEC/COMB/GOOD`.
- Se `usr1` ou `usr2` tentar se subscrever usando uma sequência de tópicos de `/SEC/COMB/BAD`, nenhum seria permitido porque não possuem acesso ao nó de tópico associado, ou nós pais desse tópico.

Uma operação de assinatura que especifica o nome de um objeto do tópico que não existe resulta em um erro de MQRC_UNKNOWN_OBJECT_NAME.

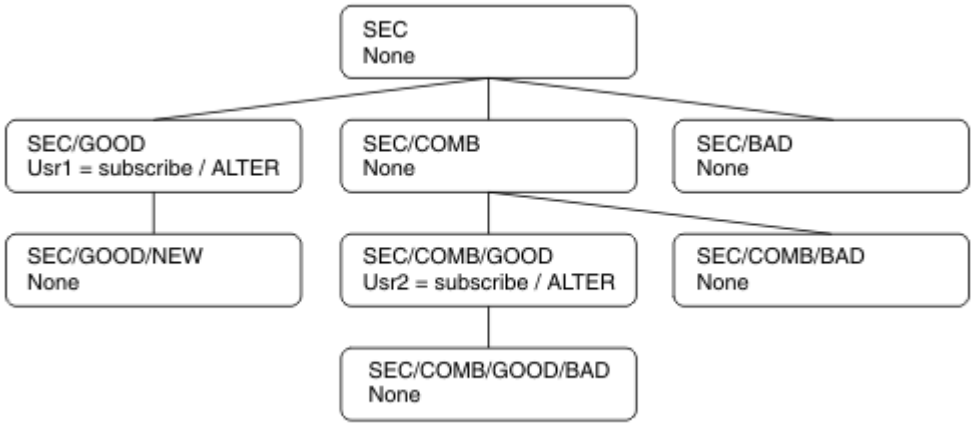
Subscrevendo Usando a Sequência de Tópicos na Qual o Nó de Tópico Existe

O comportamento é igual ao quando especificar o tópico pelo nome do objeto MQCHAR48.

Subscrevendo Usando a Sequência de Tópicos na Qual o Nó de Tópico não Existe

Considere o caso de uma assinatura de aplicativo, especificando uma sequência de tópicos que representa um nó de tópico que não existe atualmente na árvore de tópicos. A verificação de autoridade é executada conforme descrito na seção anterior. A verificação inicia com o nó pai do qual é representado pela sequência de tópicos. Se a autoridade for concedida, um novo nó que representa a sequência de tópicos será criado na árvore de tópicos.

Por exemplo, `usr1` tenta se subscrever em um tópico `SEC/GOOD/NEW`. A autoridade é concedida porque `usr1` tem acesso ao nó-pai `SEC/GOOD`. Um novo nó de tópico é criado na árvore conforme o seguinte diagrama é mostrado. O novo nó de tópico não é um objeto do tópico, ele não tem qualquer atributo de segurança associado diretamente; os atributos são herdados do seu pai.



Subscrevendo Usando uma Sequência de Tópicos que Contém Caracteres Curinga

Considere o caso de subscrever usando uma sequência de tópicos que contém um caractere curinga. A verificação de autoridade é feita no nó na árvore de tópicos que corresponde à parte completa da sequência de tópicos.

Portanto, se um aplicativo se subscrever ao SEC/COMB/GOOD/*, uma verificação de autoridade será executada conforme detalhado nas duas seções anteriores no nó SEC/COMB/GOOD na árvore de tópicos.

De maneira semelhante, se um aplicativo precisar ser subscrever ao SEC/COMB/*/GOOD, uma verificação de autoridade será executada no nó SEC/COMB.

Autoridade para Filas de Destino

Ao assinar um tópico, um dos parâmetros é o identificador do hobj de uma fila que foi aberta para saída para receber as publicações.

Se hobj não estiver especificado, mas estiver em branco, uma fila gerenciada será criada se as seguintes condições se aplicarem:

- A opção MQSO_MANAGED foi especificada.
- A assinatura não existe.
- A criação é especificada.

Se hobj estiver em branco e você estiver alterando ou continuando uma assinatura existente, a fila de destino anteriormente fornecida poderia ser gerenciada ou não gerenciada.

O aplicativo ou o usuário que faz a solicitação MQSUB deve ter a autoridade para colocar as mensagens na fila de destino que forneceu; na realidade, a autoridade para fazer com que as mensagens publicadas sejam colocadas nessa fila. A verificação de autoridade segue as regras existentes para a verificação de segurança da fila.

A verificação de segurança inclui verificações alternativas de contexto e ID do usuário onde necessário. Para conseguir configurar alguns dos campos de contexto de Identidade você deve especificar a opção MQSO_SET_IDENTITY_CONTEXT bem como a opção MQSO_CREATE ou MQSO_ALTER. Não é possível configurar qualquer um dos campos de contexto de Identidade em uma solicitação MQSO_RESUME.

Se o destino for uma fila gerenciada, nenhuma verificação de segurança será executada no destino gerenciado. Se tiver permissão para se subscrever em um tópico, é assumido você pode usar os destinos gerenciados.

Publicando Usando o Nome do Tópico ou Sequência de Tópicos Onde Existir o Nó de Tópico

O modelo de segurança para publicação é o mesmo que para assinatura, com exceção dos curingas. Publicações não contêm curingas; portanto, não há nenhum caso de uma sequência de tópicos contendo curingas para ser considerado.

As autoridades para publicar e subscrever são distintas. Um usuário ou grupo pode ter a autoridade para executar uma sem conseguir necessariamente executar a outra.

Ao publicar em um objeto do tópico especificando o nome MQCHAR48 ou a sequência de tópicos, o nó correspondente na árvore de tópicos será localizado. Se os atributos de segurança associados ao nó de tópico indicarem que o usuário tem autoridade para publicar, o acesso será concedido.

Se o acesso não for concedido, o nó pai na árvore determinará se o usuário tem autoridade para publicar nesse nível. Em caso positivo, o acesso é concedido. Em caso negativo, a recursão continuará até que seja localizado um nó que conceda autoridade de publicação ao usuário. A recursão para quando o nó-raiz é considerado sem que a autoridade tenha sido concedida. No último caso, o acesso é negado.

Em resumo, se algum nó no caminho conceder autoridade para publicar nesse usuário ou aplicativo, o publicador terá permissão para publicar nesse nó ou em qualquer lugar abaixo desse nó na árvore de tópicos.

Publicando Usando o Nome do Tópico ou Sequência de Tópicos Onde não Existir o Nó de Tópico

Assim como na operação de assinatura, quando um aplicativo é publicado, especificando uma sequência de tópicos que representa um nó de tópico que não existe atualmente na árvore de tópicos, a verificação de autoridade é executada iniciando com o pai do nó representado pela sequência de tópicos. Se a autoridade for concedida, um novo nó que representa a sequência de tópicos será criado na árvore de tópicos.

Publicando Usando uma Fila de Alias que Resolve em um Objeto de Tópico

Se você publicar usando uma fila de alias que é resolvida em um objeto de tópico, a verificação de segurança ocorrerá na fila de alias e no tópico subjacente no qual é resolvido.

A verificação de segurança na fila de alias verifica se o usuário tem autoridade para colocar as mensagens nessa fila de alias e a verificação de segurança no tópico verifica se o usuário pode publicar nesse tópico. Quando uma fila de alias é resolvida em outra fila, as verificações *não* são feitas na fila subjacente. A verificação de autoridade é executada de maneira diferente para os tópicos e as filas.

Fechando uma Assinatura

Ocorre uma verificação de segurança adicional se você fechar a assinatura usando a opção `MQCO_REMOVE_SUB` se não criar a assinatura sob esta manipulação.

Uma verificação de segurança é executada para assegurar que tenha a autoridade correta para fazer isso porque a ação resulta na remoção da assinatura. Os atributos de segurança associados ao nó de tópico indicam que o usuário tem autoridade e, em seguida, o acesso é concedido. Em caso negativo, o nó-pai na árvore é considerado para determinar se o usuário tem autoridade para fechar a assinatura. A recursão continua até que a autoridade seja concedida ou o nó-raiz seja atingido.

Definindo, Alterando e Excluindo uma Assinatura

Nenhuma verificação de segurança de assinatura é executada quando uma assinatura for criada administrativamente, em vez de usar uma solicitação de API `MQSUB`. O administrador já recebeu esta autoridade por meio do comando.

As verificações de segurança são executadas para assegurar que as publicações podem ser colocadas na fila de destino associadas à assinatura. As verificações são executadas da mesma maneira que para uma solicitação `MQSUB`.

O ID do usuário que é usado para essas verificações de segurança depende do comando sendo emitido. Se o parâmetro **SUBUSER** for especificado, ele afetará a maneira como a verificação é executada, conforme mostrado em Tabela 18 na página 262:

Tabela 18. IDs de Usuário Usados para Verificações de Segurança para Comandos

Comando:	SUBUSER especificado e em branco	SUBUSER especificado e concluído	SUBUSER não especificado
	Use o ID de administrador		Use o ID de administrador
	Use o ID de administrador		Use o ID do usuário a partir da assinatura existente

A única verificação de segurança executada ao excluir as assinaturas usando o comando DELETE SUB é a verificação de segurança de comando.

Exemplo de configuração de segurança de publicação/assinatura

Esta seção descreve um cenário que tem a configuração do controle de acesso em tópicos de uma maneira que permite que o controle de segurança a serem aplicados conforme necessário.

Conceder acesso a um usuário para assinar um tópico

Este tópico é o primeiro em uma lista de tarefas que informam como conceder acesso aos tópicos por mais de um usuário.

Sobre esta tarefa

Esta tarefa assume que nenhum objeto de tópico administrativo existe nem quaisquer perfis foram definidos para assinatura ou publicação. Os aplicativos estão criando novas assinaturas em vez de continuar as existentes e estão fazendo isso usando somente a sequência de tópicos.

Um aplicativo pode fazer uma assinatura fornecendo um objeto do tópico, uma sequência de tópicos ou uma combinação de ambos. Independentemente do caminho o aplicativo selecione, o efeito é fazer uma assinatura em um determinado ponto na árvore de tópicos. Se este ponto na árvore de tópicos for representado por um objeto de tópico administrativo, um perfil de segurança será verificado com base no nome do objeto do tópico.

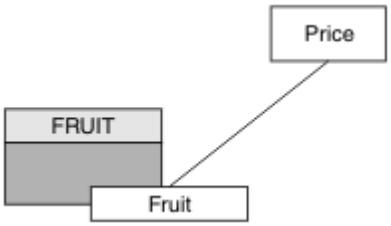


Figura 27. Exemplo de acesso do objeto do tópico

Tabela 19. Acesso ao objeto do tópico de exemplo

Tópico	Acesso de assinatura necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum
Preço/Fruta	USER1	FRUTA

Defina um novo objeto do tópico conforme a seguir:

Procedimento

1. Emita o comando MQSC `DEF TOPIC(FRUIT) TOPICSTR('Price/Fruit')`.
2. Conceda acesso como a seguir:
   - Em outras plataformas:

     Conceda acesso ao USER1 para assinar o tópico `"Price/Fruit"` , concedendo ao usuário acesso ao objeto FRUIT Faça isso usando o comando de autorização para a plataforma:

     Windows UNIX Linux **Windows, UNIX and Linux sistemas**

     ```
     setmqaut -t topic -n FRUIT -p USER1 +sub
     ```

Resultados

Quando USER1 tenta assinar o tópico `"Price/Fruit"` o resultado é bem-sucedido.

Quando USER2 tenta assinar o tópico `"Price/Fruit"` , o resultado é uma falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- Windows UNIX Linux Em outras plataformas, o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Observe que esta é uma ilustração do que você vê; não todos os campos.

Conceder acesso a um usuário para assinar um tópico mais fundo na árvore

Este tópico é o segundo em uma lista de tarefas que informam como conceder acesso aos tópicos por mais de um usuário.

Antes de começar

Este tópico usa a configuração descrita em “Conceder acesso a um usuário para assinar um tópico” na página 263.

Sobre esta tarefa

Se o ponto na árvore de tópicos no qual o aplicativo faz a assinatura não é representado por um objeto de tópico administrativo, mova para cima na árvore até que o objeto do tópico administrativo pai mais próximo seja localizado. O perfil de segurança é verificado com base no nome do objeto do tópico.

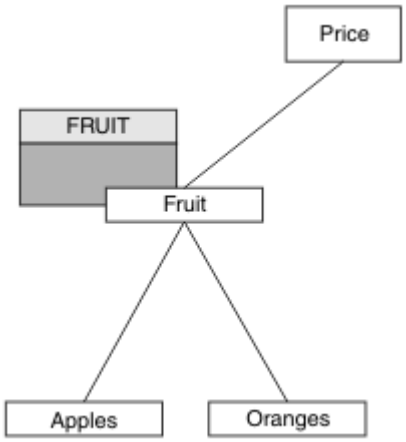


Figura 28. Exemplo de concessão de acesso a um tópico em uma árvore de tópicos

Tabela 20. Requisitos de acesso para tópicos de exemplo e objetos de tópico

Tópico	Acesso de assinatura necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum
Preço/Fruta	USER1	FRUTA
Preço/Fruta/ Maçãs	USER1	

Tabela 20. Requisitos de acesso para tópicos de exemplo e objetos de tópico (continuação)		
Tópico	**Acesso de assinatura necessário**	**Objeto do Tópico**
Preço/Fruta/ Laranjas	USER1	

Na tarefa anterior, USER1 recebeu acesso para assinar o tópico `"Price/Fruit"` concedendo a ele acesso ao perfil `hlq.SUBSCRIBE.FRUIT` no z/OS e acesso de assinatura ao perfil `FRUIT` em outras plataformas. Esse perfil único também concede ao USER1 acesso para assinar `"Price/Fruit/Apples"`, `"Price/Fruit/Oranges"` e `"Price/Fruit/#"`

Quando USER1 tenta assinar o tópico `"Price/Fruit/Apples"` o resultado é bem-sucedido.

Quando USER2 tenta assinar o tópico `"Price/Fruit/Apples"` , o resultado é uma falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- No z/OS, as mensagens a seguir vistas no console mostram o caminho de segurança completo por meio da árvore de tópicos que foi tentada:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- Em outras plataformas, o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Apples"
```

Observe o seguinte :

- As mensagens recebidas no z/OS são idênticas às recebidas na tarefa anterior, pois os mesmos objetos e perfis do tópico estão controlando o acesso.
- A mensagem do evento que você recebe em outras plataformas é semelhante à recebida na tarefa anterior, mas a sequência de tópicos real é diferente.

Garanta acesso a outro usuário para assinar somente o tópico dentro da árvore

Este tópico é o terceiro em uma lista de tarefas que informam como conceder acesso para assinar tópicos por mais de um usuário.

Antes de começar

Este tópico usa a configuração descrita em “Conceder acesso a um usuário para assinar um tópico mais fundo na árvore” na página 264.

Sobre esta tarefa

Na tarefa anterior, USER2 foi recusado o acesso ao tópico `"Price/Fruit/Apples"`. Esse tópico informa como conceder acesso a esse tópico, mas não para quaisquer outros tópicos.

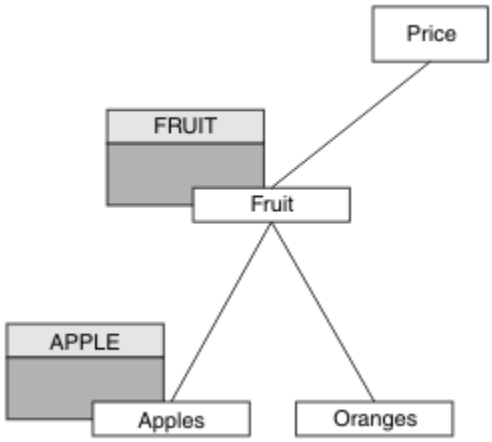


Figura 29. Concedendo acesso a tópicos específicos em uma árvore de tópicos

Tabela 21. Requisitos de acesso para tópicos de exemplo e objetos de tópico

Tópico	Acesso de assinatura necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum
Preço/Fruta	USER1	FRUTA
Preço/Fruta/ Maçãs	USER1 e USER2	APPLE
Preço/Fruta/ Laranjas	USER1	

Defina um novo objeto do tópico conforme a seguir:

Procedimento

1. Emita o comando MQSC `DEF TOPIC(APPLE) TOPICSTR('Price/Fruit/Apples')`.
2. Conceda acesso como a seguir:
   - Em outras plataformas:

     Na tarefa anterior, USER1 recebeu acesso para assinar o tópico `"Price/Fruit/Apples"` concedendo ao usuário acesso de assinatura para o perfil FRUIT .

     Esse único perfil também concedeu acesso USER1 para assinar `"Price/Fruit/Oranges"` e `"Price/Fruit/#"`e esse acesso permanece mesmo com a inclusão do novo objeto de tópico e os perfis associados.

     Conceda acesso ao USER2 para assinar o tópico `"Price/Fruit/Apples"` , concedendo ao usuário acesso de assinatura ao perfil do APPLE Faça isso usando o comando de autorização para a plataforma:

     Windows, UNIX and Linux sistemas

     ```
     setmqaut -t topic -n APPLE -p USER2 +sub
     ```

Resultados

No z/OS, quando USER1 tenta assinar o tópico `"Price/Fruit/Apples"` , a primeira verificação de segurança no perfil `hlq.SUBSCRIBE.APPLE` falha, mas ao mover para cima a árvore, o perfil

hlq.SUBSCRIBE.FRUIT permite que USER1 se inscreva, portanto, a assinatura é bem-sucedida e nenhum código de retorno é enviado para a chamada MQSUB.. No entanto, uma mensagem RACF ICH é gerada para a primeira verificação:

```
ICH408I USER(USER1   ) ...
  hlq.SUBSCRIBE.APPLE ...
```

Quando USER2 tenta assinar o tópico "Price/Fruit/Apples" , o resultado é bem-sucedido porque a verificação de segurança passa no primeiro perfil.

Quando USER2 tenta assinar o tópico "Price/Fruit/Oranges" , o resultado é uma falha com uma mensagem MQRC_NOT_AUTHORIZED , juntamente com:

- Windows UNIX Linux Nas plataformas Windows, UNIX e Linux , o seguinte evento de autorização:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

A desvantagem dessa configuração é que no z/OSvocê recebe mensagens ICH adicionais no console. É possível evitar isso se você proteger a árvore de tópicos de maneira diferente.

Mudar o controle de acesso para evitar mensagens adicionais

Este tópico é o quarto em uma lista de tarefas que informa como conceder acesso para assinar tópicos por mais de um usuário e evitar mensagens RACF ICH408I adicionais no z/OS.

Antes de começar

Este tópico aprimora a configuração descrita em “Garanta acesso a outro usuário para assinar somente o tópico dentro da árvore” na página 265 para que você evite mensagens de erro adicionais.

Sobre esta tarefa

Esse tópico informa como conceder acesso a tópicos mais profundos na árvore e como remover o acesso ao tópico para baixo na árvore quando nenhum usuário requerer isso.

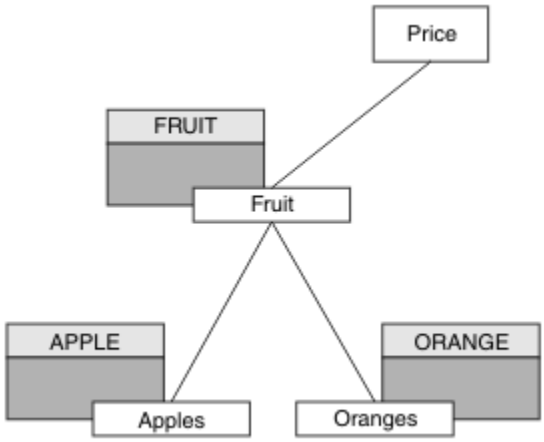


Figura 30. Exemplo de concessão do controle de acesso para evitar mensagens adicionais.

Defina um novo objeto do tópico conforme a seguir:

Procedimento

1. Emita o comando MQSC DEF TOPIC(ORANGE) TOPICSTR('Price/Fruit/Oranges').

2. Conceda acesso como a seguir:
   - Em outras plataformas:

     Configure o acesso equivalente usando os comandos de autorização para a plataforma:

     Windows UNIX Linux **Windows, UNIX and Linux sistemas**

     ```
     setmqaut -t topic -n ORANGE -p USER1 +sub
     setmqaut -t topic -n APPLE -p USER1 +sub
     ```

Resultados

No z/OS, quando USER1 tenta assinar o tópico `"Price/Fruit/Apples"` , a primeira verificação de segurança no perfil `hlq.SUBSCRIBE.APPLE` é bem-sucedida.

Da mesma forma, quando USER2 tenta assinar o tópico `"Price/Fruit/Apples"` , o resultado é bem-sucedido porque a verificação de segurança passa no primeiro perfil.

Quando USER2 tenta assinar o tópico `"Price/Fruit/Oranges"` , o resultado é uma falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- Windows UNIX Linux Em outras plataformas, o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

Conceder acesso a um usuário para publicar um tópico

Este tópico é o primeiro uma em uma lista de tarefas que informam como conceder acesso para publicar os tópicos por mais de um usuário.

Sobre esta tarefa

Esta tarefa assume que nenhum objeto de tópico administrativo exista no lado direito da árvore de tópicos, nem quaisquer perfis foram definidos para publicação. A suposição usada é que os publicadores estão usando somente a sequência de tópicos.

Um aplicativo pode publicar em um tópico fornecendo um objeto do tópico, uma sequência de tópicos ou uma combinação de ambos. Qualquer forma o aplicativo seleciona, o efeito é publicar em um determinado ponto na árvore de tópicos. Se este ponto na árvore de tópicos for representado por um objeto de tópico administrativo, um perfil de segurança será verificado com base no nome do objeto do tópico. Por exemplo:

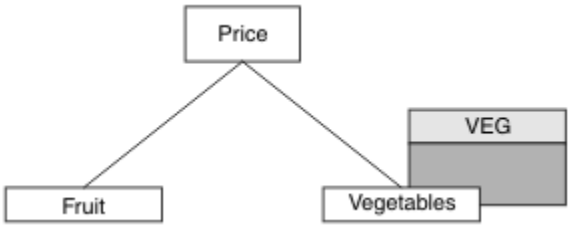


Figura 31. Concedendo acesso de publicação a um tópico

Tabela 22. Exemplo de requisitos de acesso de publicação		
Tópico	**Acesso de publicação necessário**	**Objeto do Tópico**
Preço (Real R$)	Nenhum usuário	Nenhum
Preço/Vegetais	USER1	VEG

Defina um novo objeto do tópico conforme a seguir:

Procedimento

1. Emita o comando MQSC `DEF TOPIC(VEG) TOPICSTR('Price/Vegetables')`.
2. Conceda acesso como a seguir:
   - Em outras plataformas:

     Conceda acesso ao USER1 para publicar no tópico `"Price/Vegetables"` concedendo ao usuário acesso ao perfil VEG . Faça isso usando o comando de autorização para a plataforma:

     
     Windows, UNIX and Linux sistemas

     ```
     setmqaut -t topic -n VEG -p USER1 +pub
     ```

Resultados

Quando o USER1 tenta publicar no tópico `"Price/Vegetables"` , o resultado é bem-sucedido; ou seja, a chamada MQOPEN é bem-sucedida

Quando USER2 tenta publicar no tópico `"Price/Vegetables"` , a chamada MQOPEN falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- 
  Em outras plataformas, o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables"
```

Observe que esta é uma ilustração do que você vê; não todos os campos.

Conceder acesso a um usuário para publicar em um tópico dentro da árvore

Este tópico é o segundo em uma lista de tarefas que informam como conceder acesso a publicação em tópicos por mais de um usuário.

Antes de começar

Este tópico usa a configuração descrita em “Conceder acesso a um usuário para publicar um tópico” na página 268.

Sobre esta tarefa

Se o ponto na árvore de tópicos no qual o aplicativo publicar não for representado por um objeto do tópico administrativo, mova a árvore para cima até onde o objeto do tópico administrativo pai mais próximo está localizado. O perfil de segurança é verificado com base no nome do objeto do tópico.

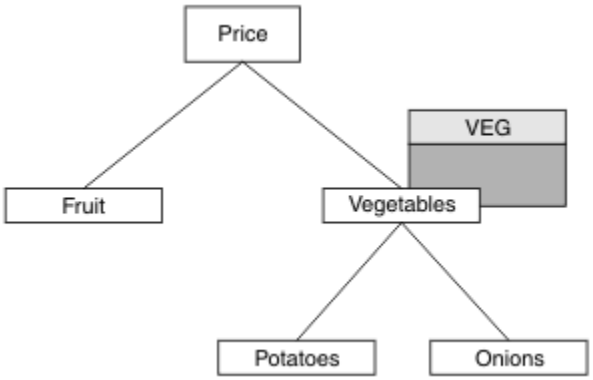


Figura 32. Concedendo acesso de publicação em um tópico em uma árvore de tópicos

Tabela 23. Exemplo de requisitos de acesso de publicação

Tópico	Acesso de assinatura necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum
Preço/Vegetais	USER1	VEG
Preço/Vegetais/ Batatas	USER1	
Preço/Vegetais/ Cebolas	USER1	

Na tarefa anterior, o USER1 recebeu acesso para publicar tópico `"Price/Vegetables/Potatoes"` concedendo a ele acesso ao perfil `hlq.PUBLISH.VEG` no z/OS ou acesso de publicação ao perfil `VEG` em outras plataformas. Esse único perfil também concede acesso USER1 para publicação em `"Price/Vegetables/Onions"`.

Quando USER1 tenta publicar no tópico `"Price/Vegetables/Potatoes"` , o resultado é bem-sucedido; essa é a chamada MQOPEN bem-sucedida.

Quando o USER2 tenta assinar o tópico `"Price/Vegetables/Potatoes"` , o resultado é falha; ou seja, a chamada MQOPEN falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- No z/OS, as mensagens a seguir vistas no console mostram o caminho de segurança completo por meio da árvore de tópicos que foi tentada:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.VEG ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- Em outras plataformas, o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables/Potatoes"
```

Observe o seguinte :

- As mensagens recebidas no z/OS são idênticas às recebidas na tarefa anterior, pois os mesmos objetos e perfis do tópico estão controlando o acesso.

- A mensagem do evento que você recebe em outras plataformas é semelhante à recebida na tarefa anterior, mas a sequência de tópicos real é diferente.

Conceder acesso para publicação e assinatura

Este tópico é o último em uma lista de tarefas que informam como conceder acesso para publicar e assinar tópicos por mais de um usuário.

Antes de começar

Este tópico usa a configuração descrita em “Conceder acesso a um usuário para publicar em um tópico dentro da árvore” na página 269.

Sobre esta tarefa

Em uma tarefa anterior, USER1 recebeu acesso para assinar o tópico `"Price/Fruit"`. Este tópico informa como conceder acesso ao usuário para publicar para o tópico.

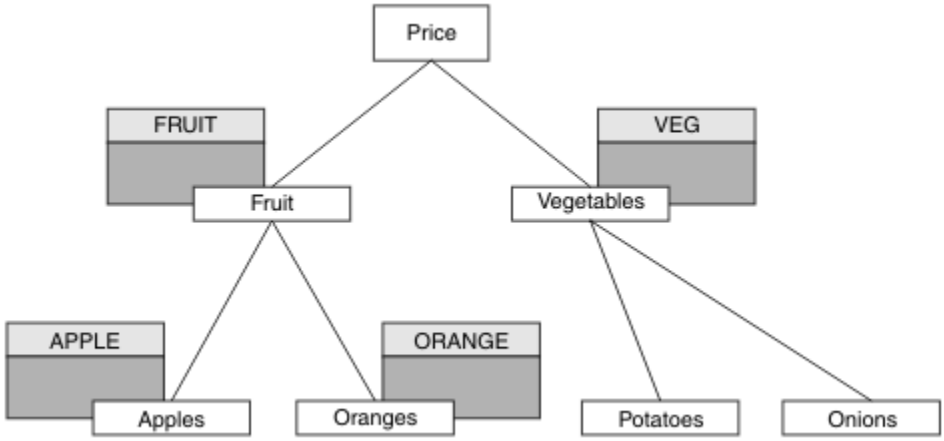


Figura 33. Concedendo acesso para publicação e assinatura

Tabela 24. Exemplo de requisitos de acesso de publicação e assinatura

Tópico	Acesso de assinatura necessário	Acesso de publicação necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum usuário	Nenhum
Preço/Fruta	USER1	USER1	FRUTA
Preço/Fruta/ Maçãs	USER1 e USER2		APPLE
Preço/Fruta/ Laranjas	USER1		LARANJA

Procedimento

Conceda acesso como a seguir:

- Em outras plataformas:

  Conceda acesso ao USER1 para publicar no tópico `"Price/Fruit"` concedendo ao usuário acesso de publicação ao perfil FRUIT . Faça isso usando o comando de autorização para a plataforma:

Windows UNIX Linux **Windows, UNIX and Linux sistemas**

```
setmqaut -t topic -n FRUIT -p USER1 +pub
```

Resultados

No z/OS, quando USER1 tenta publicar no tópico `"Price/Fruit"` , a verificação de segurança na chamada MQOPEN passa.

Quando USER2 tenta publicar no tópico `"Price/Fruit"` , o resultado é falha com uma mensagem `MQRC_NOT_AUTHORIZED` , juntamente com:

- Windows UNIX Linux Nas plataformas Windows, UNIX e Linux , o evento de autorização a seguir:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Seguindo o conjunto completo dessas tarefas, fornece USER1 e USER2 as seguintes autoridades de acesso para publicar e subscrever-se nos tópicos listados:

Tabela 25. Conclua lista de autoridades de acesso resultante de exemplos de segurança

Tópico	Acesso de assinatura necessário	Acesso de publicação necessário	Objeto do Tópico
Preço (Real R$)	Nenhum usuário	Nenhum usuário	Nenhum
Preço/Fruta	USER1	USER1	FRUTA
Preço/Fruta/ Maçãs	USER1 e USER2		APPLE
Preço/Fruta/ Laranjas	USER1		LARANJA
Preço/ Vegetais		USER1	VEG
Preço/ Vegetais/ Batatas			
Preço/ Vegetais/ Cebolas			

Quando você tem diferentes requisitos para acesso de segurança em diferentes níveis dentro da árvore de tópicos, o planejamento cuidadoso assegura que você não receba avisos de segurança externos sobre o log do console do z/OS Configurando a segurança no nível correto na árvore evita mensagens de segurança enganosas.

Segurança de assinatura

MQSO_ALTERNATE_USER_AUTHORITY

O campo AlternateUserId contém um identificador de usuário a ser usado para validar esta chamada MQSUB. A chamada pode ser bem-sucedida somente se este AlternateUserId estiver autorizado a assinar o tópico com opções de acesso especificadas, independentemente de se o ID do usuário sob o qual o aplicativo estiver em execução está autorizado para tanto.

MQSO_SET_IDENTITY_CONTEXT

A assinatura deve usar o token de conta e os dados de identificação do aplicativo fornecidos nos campos PubAccountingToken e PubApplIdentityData.

Se esta opção for especificada, a mesma verificação de autorização será executada como se a fila de destino tivesse sido acessada usando uma chamada MQOPEN com MQOO_SET_IDENTITY_CONTEXT, exceto no caso em que a opção MQSO_MANAGED também é usada; neste caso, não há nenhuma verificação de autorização na fila de destino.

Se esta opção não for especificada, as publicações enviadas a este assinante terão informações de contexto padrão associadas a elas da seguinte maneira:

Tabela 26. Informações de contexto de publicação padrão

Campo no MQMD	**Valor Usado**
UserIdentifier	O ID do usuário associado à assinatura (consulte o campo SUBUSER em DISPLAY SBSTATUS) no momento em que a publicação é feita.
AccountingToken	Determinado a partir do ambiente, se possível; caso contrário, configure como MQACT_NONE.
ApplIdentityData	Configure como em branco.

Esta opção é válida apenas com MQSO_CREATE e MQSO_ALTER. Se usada com MQSO_RESUME, os campos PubAccountingToken e PubApplIdentityData são ignorados, portanto, esta opção não tem efeito.

Se uma assinatura é alterada sem o uso dessa opção na qual a opção forneceu informações de contexto anteriormente, as informações de contexto padrão são geradas para assinatura alterada.

Se uma assinatura permitindo que diferentes IDs de usuário a usem com a opção MQSO_ANY_USERID for continuada por um ID do usuário diferente, o contexto de identidade padrão será gerado para o novo ID do usuário que agora possui a assinatura e todas as publicações subsequentes serão entregues contendo o novo contexto de identidade.

AlternateSecurityId

Este é um identificador de segurança que é transmitido com o AlternateUserId para o serviço de autorização para permitir que verificações de autorização apropriadas sejam executadas. AlternateSecurityId é usado somente se MQSO_ALTERNATE_USER_AUTHORITY for especificado e o campo AlternateUserId não estiver completamente em branco até o primeiro caractere nulo ou no final do campo.

Opção de Assinatura MQSO_ANY_USERID

Quando MQSO_ANY_USERID é especificado, a identidade do assinante não é restrita a um único ID do usuário. Isso permite que qualquer usuário altere ou continue a assinatura quando tem autoridade adequada. Apenas um único usuário pode ter a assinatura a qualquer momento. Uma tentativa de continuar o uso de uma assinatura atualmente em uso por outro aplicativo irá fazer com que a chamada falhe com MQRC_SUBSCRIPTION_IN_USE.

Para incluir essa opção a uma assinatura existente a chamada MQSUB (utilizando MQSO_ALTER) deve vir do mesmo ID do usuário da assinatura original.

Se uma chamada MQSUB se referir a uma assinatura existente com MQSO_ANY_USERID configurado e o ID do usuário for diferente da assinatura original, a chamada será bem-sucedida apenas se o novo ID do usuário tiver autoridade para assinar o tópico. Após concluir com sucesso, publicações futuras a este assinante serão colocadas na fila do assinante com o novo ID do usuário configurado na publicação.

MQSO_FIXED_USERID

Quando MQSO_FIXED_USERID é especificado, a assinatura somente pode ser alterada ou retomada por um único ID do usuário proprietário. Este ID do usuário é o último ID do usuário a alterar a assinatura que configura esta opção, removendo a opção MQSO_ANY_USERID ou se nenhuma alteração ocorreu, é o ID do usuário que criou a assinatura.

Se um verbo MQSUB se referir a uma assinatura existente com MQSO_ANY_USERID configurado e altera a assinatura (usando MQSO_ALTER) para usar a opção MQSO_FIXED_USERID, o ID do usuário da assinatura agora será fixo nesse novo ID do usuário. A chamada será bem-sucedida apenas se o novo ID do usuário tiver autoridade para assinar o tópico.

Se um ID do usuário diferente do registrado como pertencente a uma assinatura tenta continuar ou alterar uma assinatura MQSO_FIXED_USERID, a chamada falhará com MQRC_IDENTITY_MISMATCH. O ID do usuário proprietário de uma assinatura pode ser visualizado usando o comando DISPLAY SBSTATUS.

Se nem MQSO_ANY_USERID ou MQSO_FIXED_USERID forem especificados, o padrão será MQSO_FIXED_USERID.

IBM WebSphere MQ Advanced Message Security

IBM WebSphere MQ Advanced Message Security (AMS) é um componente licenciado separadamente do IBM WebSphere MQ Advanced Message Security que fornece um alto nível de proteção para dados sensíveis que fluem pela rede do IBM WebSphere MQ Advanced Message Security , enquanto não impacta os aplicativos finais.

Visão geral do IBM WebSphere MQ Advanced Message Security

Os aplicativos do IBM WebSphere MQ podem usar IBM WebSphere MQ Advanced Message Security para enviar dados confidenciais, como transações financeiras de valor alto e informações pessoais, com níveis diferentes de proteção usando um modelo de criptografia de chave pública.

Referências relacionadas

Códigos de retorno do GSKit usados em mensagens do IBM WebSphere MQ AMS

Comportamento que mudou entre a versão 7.0.1 e a versão 7.5

Como o IBM Advanced Message Security se tornou um componente no WebSphere MQ 7.5, alguns aspectos da funcionalidade do IBM WebSphere MQ AMS foram alterados, o que pode afetar os aplicativos existentes, scripts administrativos ou procedimentos de gerenciamento

Revise a lista a seguir de mudanças cuidadosamente antes de atualizar os gerenciadores de filas para a versão 7.5. Decida se você deve planejar fazer mudanças em aplicativos, scripts e procedimentos existentes antes de iniciar a migração de sistemas para IBM WebSphere MQ versão 7.5:

- A instalação do IBM WebSphere MQ AMS faz parte do processo de instalação do WebSphere MQ.
- recursos de segurança do IBM WebSphere MQ AMS são ativados com sua instalação e controlados com políticas de segurança. Não é necessário ativar interceptores para permitir que o IBM WebSphere MQ AMS comece a interceptar dados.
- O IBM WebSphere MQ AMS na WebSphere MQ versão 7.5 não requer o uso do comando **cfgmqs** como na versão independente do IBM WebSphere MQ AMS

Características e funções do IBM WebSphere MQ Advanced Message Security

O Advanced Message Security expande serviços de segurança do WebSphere MQ para fornecer dados de assinatura e criptografia no nível de mensagem. Os serviços expandidos garantem que os dados da mensagem não foram modificados entre quando foram originalmente colocados em uma fila e quando foram recuperados. Além disso, o IBM WebSphere MQ AMS verifica se um emissor de dados da mensagem está autorizado a colocar mensagens assinadas em uma fila de destino.

Aqui está uma lista completa de funções do IBM WebSphere MQ AMS :

- Protege transações sensíveis ou de valor alto processadas pelo WebSphere MQ.
- Detecta e remove mensagens prejudiciais ou não autorizadas antes de serem processadas por uma aplicação de recepção.
- Verifica se as mensagens não foram modificadas durante a passagem de uma fila para outra.
- Protege os dados não só à medida que eles fluem através da rede, mas também quando são colocados em uma fila.
- Protege os aplicativos existentes de propriedade e gravados pelo cliente para o WebSphere MQ.

Manipulação de Erros

O Advanced Message Security define uma fila de manipulação de erros para gerenciar mensagens que contêm erros ou mensagens que não podem ser desprotegidas.

As mensagens defeituosas são tratadas como casos excepcionais. Se uma mensagem recebida não atender aos requisitos de segurança para a fila em que ela está, por exemplo, se a mensagem estiver assinada quando deveria estar criptografada ou a decriptografia ou verificação de assinatura falhar, a mensagem será enviada para a fila de manipulação de erros. Uma mensagem pode ser enviada para a fila de manipulação de erros pelas razões a seguir:

- Incompatibilidade de quality of protection - uma incompatibilidade de quality of protection (QOP) existe entre a mensagem recebida e a definição de QOP na política de segurança.
- Erro de decriptografia - a mensagem não pode ser decriptografada.
- Erro de cabeçalho PDMQ-o cabeçalho da mensagem AMS do WebSphere MQ não pode ser acessado
- Incompatibilidade de tamanho - comprimento de uma mensagem após a decriptografia ser diferente daquela esperada.
- Incompatibilidade de intensidade de algoritmo de criptografia - o algoritmo de criptografia de mensagem é mais fraco do que o necessário.
- Erro desconhecido - ocorreu um erro inesperado.

O WebSphere MQ AMS usa o SYSTEM.PROTECTION.ERROR.QUEUE como sua fila de manipulação de erros.. Todas as mensagens colocadas pelo IBM WebSphere MQ AMS no sistema SYSTEM.PROTECTION.ERROR.QUEUE são precedidos pelo cabeçalho MQDLH

O administrador do WebSphere MQ também pode definir o SYSTEM SYSTEM.PROTECTION.ERROR.QUEUE como uma fila de alias apontando para outra fila.

Conceitos chave

Aprenda sobre os conceitos chave no Advanced Message Security para entender como a ferramenta funciona e como gerenciar de forma efetiva.

Infraestrutura de chave pública

A infraestrutura de chave pública (PKI) é um sistema de recursos, políticas e serviços que suportam o uso de criptografia de chave pública para obter comunicação segura.

Não há um padrão que defina os componentes de uma infraestrutura de chave pública, mas um PKI geralmente envolve o uso de certificados de chave pública e inclui autoridades de certificação (CA) e outras autoridades de registro (RA) que fornecem os serviços a seguir:

- Emissão de certificados digitais
- Validação de certificados digitais
- Revogação de certificados digitais
- Distribuindo Certificados

A identidade de usuários e aplicativos é representada pelo campo **nome distinto (DN)** em um certificado associado a mensagens assinadas ou criptografadas. O Advanced Message Security usa essa identidade para representar um usuário ou um aplicativo Para autenticar essa identidade, o usuário ou aplicativo deve ter acesso ao armazenamento de chaves no qual o certificado e a chave privada associada são armazenados. Cada certificado é representado por um rótulo no keystore.

Conceitos relacionados

"Usando keystores e certificados" na página 299
Para fornecer proteção criptográfica transparente para aplicativos WebSphere MQ, o Advanced Message Security usa o arquivo keystore, em que certificados de chave pública e uma chave privada são armazenados.

Certificados Digitais

O Advanced Message Security associa usuários e aplicativos com certificados digitais padrão X.509. Certificados X.509 são geralmente assinados por uma autoridade de certificação (CA) confiável e envolvem as chaves públicas e privadas que são usadas para criptografia e decriptografia.

Os certificados digitais fornecem proteção contra identidades pela ligação de uma chave pública a seu proprietário, se esse for um indivíduo, um gerenciador de filas ou alguma outra entidade. Certificados digitais também são conhecidos como certificados de chave pública, porque eles oferecem a garantia sobre a propriedade de uma chave pública quando você usa um esquema de chave assimétrica. Este esquema requer que uma chave pública e uma chave privada sejam geradas para um aplicativo. Os dados criptografados com a chave pública só podem ser decriptografados usando a chave privada correspondente enquanto os dados criptografados com a chave privada só podem ser decriptografados usando a chave pública correspondente. A chave privada é armazenada em um arquivo do banco de dados de chaves protegido por senha. Apenas o proprietário tem acesso à chave privada usada para decriptografar mensagens que foram criptografadas usando a chave pública correspondente.

Se chaves públicas forem enviadas diretamente por seu proprietário a outra entidade, há um risco de que a mensagem possa ser interceptada e a chave pública ser substituída por outra. Isso é conhecido como um ataque "man-in-the-middle". A solução é trocar chaves públicas por meio de terceiros confiáveis, fornecendo ao usuário uma garantia segura de que a chave pública pertence à entidade com a qual você está se comunicando. Em vez de enviar sua chave pública diretamente, peça aos terceiros confiáveis para incorporá-la a um certificado digital. Os terceiros confiáveis que emitem certificados digitais são chamados de autoridade de certificação (CA).

Para obter mais informações sobre certificados digitais, consulte O que é um certificado digital.

Um certificado digital contém a chave pública de uma entidade e afirma que a chave pública pertence àquela entidade:

- quando um certificado for para uma entidade individual, ele será chamado de *certificado pessoal* ou *certificado de usuário*.
- quando um certificado for para uma autoridade de certificação, ele será chamado de *certificado CA* ou *certificado de assinante*.

Nota: O Advanced Message Security suporta certificados autoassinados em aplicativos Java e nativos

Conceitos relacionados

"Criptografia" na página 7
Criptografia é o processo de conversão entre texto legível, chamado *texto simples*, e uma forma ilegível, chamada *texto cifrado*.

Gerenciador de autoridade de objeto

O Object Authority Manager (OAM) é o componente de serviço de autorização fornecido com os produtos WebSphere MQ .

O acesso às entidades Advanced Message Security é controlado por meio de grupos de WebSphere MQ do usuário e do OAM. Os administradores podem usar a interface da linha de comandos para conceder ou revogar as autorizações, conforme necessário. Diferentes grupos de usuários podem ter diferentes tipos de autoridade de acesso para os mesmos objetos. Por exemplo, um grupo pode executar ambas as operações PUT e GET para uma fila específica enquanto outro grupo pode ter permissão somente para navegar na fila. Da mesma forma, alguns grupos podem ter a autoridade GET e PUT a para uma fila, mas não têm permissão para alterar ou excluir a fila.

Através do OAM, é possível controlar:

- O acesso a objetos do Advanced Message Security por meio do MQI. Quando um programa de aplicativo tenta acessar objetos, o OAM verifica se o perfil do usuário fazendo a solicitação tem a autorização para a operação solicitada. Isso significa que as filas e as mensagens nas filas podem ser protegidas contra acesso não autorizado.
- Permissão de usar comandos PCF e MQSC.

Conceitos relacionados

Gerenciador de autoridade de objeto

Tecnologia suportada

O Advanced Message Security depende de vários componentes de tecnologia para fornecer uma infraestrutura de segurança.

O Advanced Message Security suporta as interfaces de programação de aplicativos (APIs) do WebSphere MQ a seguir:

- MQI (Message Queue Interface)
- WebSphere MQ Java Message Service (JMS) 1.0.2 e 1.1.
- WebSphere MQ Classes Base para Java
- Classes do WebSphere MQ para .Net em um modo não gerenciado

Nota: O Advanced Message Security suporta autoridades de certificado compatíveis com X.509.

Limitações conhecidas

Aprenda sobre limitações do IBM WebSphere MQ Advanced Message Security.

- As opções IBM WebSphere MQ a seguir não são suportadas:
  - Publicação/assinatura.
  - Conversão de dados do canal.
  - Listas de distribuição.
  - Segmentação da mensagem do aplicativo
  - O uso de aplicativos não encadeados usando saída da API em plataformas HP-UX.
  - Classes IBM WebSphere MQ para .NET em um modo gerenciado (conexões de cliente ou ligações).
  - Cliente Message Service para aplicativos .NET (XMS).
  - Cliente do serviço de mensagens para aplicativos C/C++ (IA94 supportPac XMS).
- Todos os aplicativos Java são dependentes do IBM Java Runtime.

  IBM WebSphere MQ Advanced Message Security não suporta JRE fornecido por outros fornecedores.
- Aplicativos clientes JMS e Java usando IBM WebSphere MQ Advanced Message Security no modo de cliente

Qualquer aplicativo cliente JMS ou Java(incluindo agentes IBM WebSphere MQ Explorer e IBM WebSphere MQ Managed File Transfer) não pode usar IBM WebSphere MQ Advanced Message Security no modo cliente com um WebSphere MQ gerenciador de filas anterior a Version 7.5.

Para usar políticas de proteção de mensagens, esses aplicativos precisam interagir com um gerenciador de fila do IBM WebSphere MQ Version 7.5 ou se conectar no modo de ligações locais a um gerenciador de fila na mesma máquina do aplicativo.

- Você deve evitar colocar dois ou mais certificados com os mesmos Nomes Distintos, em um único arquivo keystore, porque o funcionamento do intereceptor IBM WebSphere MQ Advanced Message Security com esses certificados é indefinido.
- O adaptador de recursos IBM WebSphere MQ Version 7.5 não suporta IBM WebSphere MQ Advanced Message Security. Se a proteção de mensagens precisar ser usada com aplicativos IBM WebSphere MQ classes for JMS ou IBM WebSphere MQ classes for Java em execução em um ambiente de servidor de aplicativos, então:
  - O servidor de aplicativos deve ser configurado para usar o adaptador de recursos Version 8.0 ou posterior.
  - Ou a intercepção do MCA (Message Channel Agent) deve ser usada.

Cenários do usuário

Familiarize-se com cenários possíveis para entender quais metas de negócios é possível obter com o Advanced Message Security.

Guia de Iniciação Rápida para Plataformas Windows

Use este guia para configurar rapidamente o IBM Advanced Message Security para fornecer segurança de mensagem em plataformas Windows No momento em que você concluir, você terá criado um banco de dados de chaves para verificar identidades do usuário e definido políticas de assinatura/criptografia para o seu gerenciador de filas.

Antes de começar

É necessário ter pelo menos os recursos a seguir instalados em seu sistema:

- Servidor
- Kit de ferramentas de desenvolvimento (para os programas de amostra)
- Advanced Message Security

Consulte os recursos do IBM WebSphere MQ para os sistemas Windows para obter detalhes.

Para obter informações sobre como usar o comando **setmqenv** para inicializar o ambiente atual de modo que os comandos apropriados do WebSphere MQ possam ser localizados e executados pelo sistema operacional, consulte setmqenv

1. Criando um gerenciador de filas e uma fila

Sobre esta tarefa

Todos os exemplos a seguir usam uma fila nomeada TEST.Q para passar mensagens entre aplicativos. Advanced Message Security usa interceptores para assinar e criptografar mensagens no ponto em que elas entram na infraestrutura do WebSphere MQ por meio da interface padrão do WebSphere MQ A configuração básica é feita no WebSphere MQ e é configurada nas etapas a seguir.

É possível usar o WebSphere MQ Explorer para criar o gerenciador de filas QM_VERIFY_AMS e sua fila local chamada TEST.Q usando todas as configurações do assistente padrão ou é possível usar os comandos localizados em \WebSphere MQ\bin Lembre-se de que deve-se ser um membro do grupo de usuários mqm para executar os comandos administrativos a seguir.

Procedimento

1. Crie um gerenciador de filas

```
crtmqm QM_VERIFY_AMS
```

2. Iniciar o Gerenciador de Filas

```
strmqm QM_VERIFY_AMS
```

3. Crie uma fila chamada TEST.Q inserindo o comando a seguir em **runmqsc** para o gerenciador de filas QM_VERIFY_AMS

```
DEFINE QLOCAL(TEST.Q)
```

Resultados

Se o procedimento for concluído, o comando inserido em **runmqsc** irá exibir detalhes sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Criando e autorizando usuários

Sobre esta tarefa

Há dois usuários que aparecem neste exemplo: alice, o emissor e bob, o receptor. Para usar a fila do aplicativo, esses usuários precisam receber autoridade para usá-la. Além disso, para usar com sucesso as políticas de proteção que vamos definir estes usuários devem ser concedidos acesso a algumas filas do sistema. Para obter mais informações sobre o comando **setmqaut**, consulte **setmqaut** .

Procedimento

1. Crie os dois usuários e assegure que HOMEPATH e HOMEDRIVE estejam configurados para ambos esses usuários.
2. Autorize os usuários para se conectarem ao gerenciador de filas e para trabalhar com a fila

```
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. Também deve-se permitir que os dois usuários naveguem na fila de política do sistema e coloquem mensagens na fila de erros.

```
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
```

Resultados

Os usuários agora são criados e as autoridades necessárias concedidas a eles.

Como proceder a seguir

Para verificar se as etapas foram realizadas corretamente, use as amostras amqsput e amqsget conforme descrito na seção “7. Testando a configuração” na página 282.

3. Criando banco de dados de chaves e certificados

Sobre esta tarefa

O interceptor requer a chave pública dos usuários de envio para criptografar a mensagem. Portanto, o banco de dados de chaves das identidades do usuário mapeado para chaves pública e privada deve ser criado. No sistema real, no qual os usuários e aplicativos são dispersos por meio de vários computadores,

cada usuário teria seu próprio keystore privado. Da mesma forma, nesse guia, criamos banco de dados de chaves para `alice` e bob e compartilhamos os certificados de usuário entre eles.

Nota: Neste guia, usamos os aplicativos de amostra escritos em C conectando usando ligações locais. Se você planeja usar aplicativos Java usando ligações do cliente, deverá criar um keystore JKS e certificados usando o comando **keytool** , que faz parte do JRE (consulte "Guia de iniciação rápida para clientes Java" na página 289 para obter mais detalhes). Para todos os outros idiomas e para aplicativos Java usando ligações locais, as etapas neste guia estão corretas.

Procedimento

1. Use a GUI do IBM Key Management (`strmqikm.exe`) para criar um novo banco de dados de chaves para o usuário `alice`.

   ```
   Type:       CMS
   Filename:   alicekey.kdb
   Location:   C:/Documents and Settings/alice/AMS
   ```

   Nota:

   - É aconselhável usar uma senha forte para proteger o banco de dados.
   - Certifique-se de que a caixa de seleção **armazenar a senha em arquivo stash** foi selecionada.
2. Mude a visualização de conteúdo do banco de dados chave para **Certificados pessoais**.
3. Selecione **Novo autoassinado**; os certificados autoassinados são usados neste cenário.
4. Crie um certificado identificando o usuário `alice` para uso na criptografia usando estes campos:

   ```
   Key label: Alice_Cert
   Common Name: alice
   Organisation: IBM
   Country: GB
   ```

   Nota:

   - Para o propósito desse guia, estamos usando certificado autoassinado que pode ser criado sem o uso de uma Autoridade de Certificação. Para sistemas de produção, é aconselhável não usar certificados autoassinados, mas em vez disso contar com certificados assinados por uma Autoridade de certificação.
   - O parâmetro **`Key label`** especifica o nome para o certificado, que os interceptores procurarão para receber informações necessárias.
   - O **`Common Name`** e parâmetros opcionais especificam os detalhes do **Nome distinto** (DN), que deve ser exclusivo para cada usuário.
5. Repita as etapas 1-4 para o usuário bob

Resultados

Os dois usuários `alice` e bob agora possuem cada um certificado autoassinado.

4. Criando keystore.conf

Sobre esta tarefa

Deve-se apontar os interceptores do Advanced Message Security para o diretório no qual os bancos de dados de chaves e os certificados estão located.This é feito por meio do arquivo `keystore.conf` , que contém essas informações em formato de texto simples. Cada usuário deve ter um arquivo `keystore.conf` separado. Esta etapa deve ser feita para ambos, `alice` e bob.

O conteúdo de `keystore.conf` deve ter o formato:

```
cms.keystore = <dir>/keystore_file
cms.certificate = certificate_label
```

Exemplo

Para este cenário, o conteúdo do `keystore.conf` será o seguinte:

```
cms.keystore = C:/Documents and Settings/alice/AMS/alicekey
cms.certificate = Alice_Cert
```

Nota:

- O caminho para o arquivo keystore deve ser fornecido sem extensão de arquivo.
- O rótulo certificado pode incluir espaços, portanto, "Alice_Cert" e "Alice_Cert" por exemplo, são reconhecidas como etiquetas de dois certificados diferentes No entanto, para evitar confusão, é melhor não usar espaços no nome do rótulo.
- Há os seguintes formatos de armazenamento de chaves: CMS (Cryptographic Message Syntax), JKS (Java Keystore) e JCEKS (Java Cryptographic Extension Keystore). Para obter informações adicionais, consulte “Estrutura do arquivo de configuração do keystore (keystore.conf)” na página 300.
- *%HOMEDRIVE%\%HOMEPATH%*`\.mqs\keystore.conf` (por exemplo, C:\Documents and Settings\alice\.mqs\keystore.conf) é o local padrão em que o Advanced Message Security procura o arquivo `keystore.conf`. Para obter informações sobre como usar um local não padrão para o `keystore.conf`, consulte “Usando keystores e certificados” na página 299.
- Para criar o diretório `.mqs`, deve-se usar o prompt de comandos.

5. Compartilhando os certificados

Sobre esta tarefa

Compartilhe os certificados entre os dois bancos de dados de chaves para que cada usuário possa identificar com sucesso o outro. Isso é feito extraindo cada certificado público do usuário para um arquivo, que é, então, incluído no banco de dados de chaves do outro usuário.

Nota: Tome cuidado para usar a opção *extract* e não a opção *export*. *Extract* obtém a chave pública do usuário, enquanto *export* obtém a chave pública e a privada. Usar *export* por engano comprometeria completamente o seu aplicativo, transmitindo a sua chave privada.

Procedimento

1. Extraia o certificado identificando `alice` para um arquivo externo:

   ```
   runmqakm -cert -extract -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
   -label Alice_Cert -target alice_public.arm
   ```

2. Inclua o certificado para o keystore bob's:

   ```
   runmqakm -cert -add -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd -label
   Alice_Cert -file alice_public.arm
   ```

3. Repita as etapas para bob:

   ```
   runmqakm -cert -extract -db "C:/Documents and Settings/alice/AMS/bobkey.kdb" -pw passw0rd
   -label Bob_Cert -target bob_public.arm

   runmqakm -cert -add -db "C:/Documents and Settings/bob/AMS/alicekey.kdb" -pw passw0rd -label
   Bob_Cert -file bob_public.arm
   ```

Resultados

Os dois usuários, `alice` e bob, agora podem identificar um ao outro com êxito como tendo criado e compartilhado certificados autoassinados.

Como proceder a seguir

Verifique se um certificado está no keystore procurando por ele usando a GUI ou executando os comandos a seguir que imprimem seus detalhes:

```
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. Definindo a política de filas

Sobre esta tarefa

Com o gerenciador de filas criado e os interceptores preparados para interceptar mensagens e acessar chaves de criptografia, podemos começar a definir políticas de proteção no QM_VERIFY_AMS usado o comando `setmqspl`. Consulte setmqspl para obter mais informações sobre este comando. Cada nome da política deve ser o mesmo que o nome da fila para a qual ela deve ser aplicada.

Exemplo

Este é um exemplo de uma política definida para a fila TEST.Q. No exemplo, as mensagens são assinadas com o algoritmo SHA1 e criptografados com o algoritmo AES256. `alice` é o único emissor válido e bob é o único receptor das mensagens nesta fila:

```
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"
```

Nota: Os DNs correspondem exatamente aos especificados no respectivo certificado do usuário do banco de dados de chaves.

Como proceder a seguir

Para verificar a política que você definiu, emita o comando a seguir:

```
dspmqspl -m QM_VERIFY_AMS
```

Para imprimir os detalhes da política como um conjunto de comandos `setmqspl`, o sinalizador `-export`. Isso permite armazenar políticas já definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Testando a configuração

Sobre esta tarefa

Ao executar programas diferentes em diferentes usuários é possível verificar se o aplicativo foi configurado corretamente.

Procedimento

1. Alterne o usuário para ser executado como `alice`

   Clique com o botão direito em `cmd.exe` e selecione **Executar como...**. Quando solicitado, efetue login como o usuário `alice`.
2. Como o usuário `alice` coloque uma mensagem usando um aplicativo de amostra:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```
3. Digite o texto da mensagem e em seguida pressione Enter.
4. Alterne o usuário para ser executado como bob

   Abra outra janela clicando com o botão direito em `cmd.exe` e selecionando **Executar como...**. Quando solicitado, efetue login como o usuário bob.
5. Como o usuário Bob obtém uma mensagem usando um aplicativo de amostra:

```
amqsget TEST.Q QM_VERIFY_AMS
```

Resultados

Se o aplicativo foi configurado corretamente para ambos os usuários, a mensagem do usuário `alice`será exibida quando bob executar o aplicativo de obtenção.

8. Testando a criptografia

Sobre esta tarefa

Para verificar se a criptografia está ocorrendo conforme o esperado, crie uma fila de alias que faça referência à fila original TEST.Q. Esta fila de alias não terá uma política de segurança e, portanto, nenhum usuário terá as informações para decriptografar a mensagem e, portanto, os dados criptografados serão mostrados.

Procedimento

1. Usando o comando **runmqsc** no gerenciador de filas QM_VERIFY_AMS, crie uma fila de alias.

   ```
   DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
   ```

2. Conceda a bob acesso para procurar da fila de alias

   ```
   setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
   ```

3. Como o usuário `alice`, coloque outra mensagem usando um aplicativo de amostra como antes:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```

4. Como o usuário bob, procure a mensagem usando um aplicativo de amostra por meio da fila de alias dessa vez:

   ```
   amqsbcg TEST.ALIAS QM_VERIFY_AMS
   ```

5. Como o usuário bob, obtenha a mensagem usando um aplicativo de amostra a partir da fila local:

   ```
   amqsget TEST.Q QM_VERIFY_AMS
   ```

Resultados

A saída do aplicativo `amqsbcg` mostra os dados criptografados que estão na fila, provando que a mensagem foi criptografada.

Guia de Iniciação Rápida para Plataformas UNIX

Use este guia para configurar rapidamente IBM Advanced Message Security para fornecer segurança de mensagem em plataformas UNIX . No momento em que você concluir, você terá criado um banco de dados de chaves para verificar identidades do usuário e definido políticas de assinatura/criptografia para o seu gerenciador de filas.

Antes de começar

É necessário ter pelo menos os componentes a seguir instalados em seu sistema:

- Tempo de execução
- Servidor
- Programas de Amostra
- IBM Kit de Segurança Global
- MQ Advanced Message Security

Consulte os tópicos a seguir para os nomes dos componentes em cada plataforma específica:

- IBM WebSphere MQ componentes para Linux sistemas
- IBM WebSphere MQ componentes para HP-UX sistemas
- IBM WebSphere MQ componentes para AIX sistemas
- IBM WebSphere MQ componentes para Solaris sistemas

1. Criando um gerenciador de filas e uma fila

Sobre esta tarefa

Todos os exemplos a seguir usam uma fila nomeada TEST.Q para passar mensagens entre aplicativos. Advanced Message Security usa interceptores para assinar e criptografar mensagens no ponto em que elas entram na infraestrutura do WebSphere MQ por meio da interface padrão do WebSphere MQ A configuração básica é feita no WebSphere MQ e é configurada nas etapas a seguir.

É possível usar o WebSphere MQ Explorer para criar o gerenciador de filas QM_VERIFY_AMS e sua fila local chamada TEST.Q usando todas as configurações do assistente padrão ou é possível usar os comandos localizados em <MQ_INSTALL_PATH>/bin Lembre-se de que deve-se ser um membro do grupo de usuários mqm para executar os comandos administrativos a seguir.

Procedimento

1. Crie um gerenciador de filas

   ```
   crtmqm QM_VERIFY_AMS
   ```

2. Iniciar o Gerenciador de Filas

   ```
   strmqm QM_VERIFY_AMS
   ```

3. Crie uma fila chamada TEST.Q inserindo o comando a seguir em **runmqsc** para o gerenciador de filas QM_VERIFY_AMS

   ```
   DEFINE QLOCAL(TEST.Q)
   ```

Resultados

Se o procedimento foi concluído com sucesso, o comando a seguir inserido em **runmqsc** irá exibir detalhes sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Criando e autorizando usuários

Sobre esta tarefa

Há dois usuários que aparecem neste exemplo: alice, o emissor e bob, o receptor. Para usar a fila do aplicativo, esses usuários precisam receber autoridade para usá-la. Além disso, para usar com sucesso as políticas de proteção que vamos definir estes usuários devem ser concedidos acesso a algumas filas do sistema. Para obter mais informações sobre o comando **setmqaut** , consulte **setmqaut** .

Procedimento

1. Crie os dois usuários

   ```
   useradd alice
   useradd bob
   ```

2. Autorize os usuários para se conectarem ao gerenciador de filas e para trabalhar com a fila

   ```
   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. Também deve-se permitir que os dois usuários naveguem na fila de política do sistema e coloquem mensagens na fila de erros.

```
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
```

Resultados

Agora os grupos de usuários são criados e as autoridades requeridas concedidas a eles. Desse modo os usuários que estiverem designados a esses grupos também terão permissão para se conectar ao gerenciador de filas e colocar e obter da fila.

Como proceder a seguir

Para verificar se as etapas foram realizadas corretamente, use as amostras `amqsput` e `amqsget` conforme descrito na seção "8. Testando a criptografia" na página 288.

3. Criando banco de dados de chaves e certificados

Sobre esta tarefa

Para criptografar a mensagem, o interceptor requer a chave privada do usuário de envio e a chave pública(s) do destinatário(s). Portanto, o banco de dados de chaves das identidades do usuário mapeado para chaves pública e privada deve ser criado. No sistema real, no qual os usuários e aplicativos são dispersos por meio de vários computadores, cada usuário teria seu próprio keystore privado. Da mesma forma, nesse guia, criamos banco de dados de chaves para `alice` e bob e compartilhamos os certificados de usuário entre eles.

Nota: Neste guia, usamos os aplicativos de amostra escritos em C conectando usando ligações locais. Se você planeja usar aplicativos Java usando ligações do cliente, deverá criar um keystore JKS e certificados usando o comando **`keytool`** , que faz parte do JRE (consulte "Guia de iniciação rápida para clientes Java" na página 289 para obter mais detalhes). Para todos os outros idiomas e para aplicativos Java usando ligações locais, as etapas neste guia estão corretas.

Procedimento

1. Crie um novo banco de dados de chaves para o usuário `alice`

```
mkdir /home/alice/.mqs -p
runmqakm -keydb -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -stash
```

Nota:

- É aconselhável usar uma senha forte para proteger o banco de dados.
- O parâmetro `stash` armazena a senha no arquivo `key.sth`, que interceptores podem usar para abrir o banco de dados.

2. Assegure que o banco de dados chave é legível

```
chmod +r /home/alice/.mqs/alicekey.kdb
```

3. Crie um certificado que identifica o usuário `alice` para uso na criptografia

```
runmqakm -cert -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd
-label Alice_Cert -dn "cn=alice,o=IBM,c=GB" -default_cert yes
```

Nota:

- Para o propósito desse guia, estamos usando certificado autoassinado que pode ser criado sem o uso de uma Autoridade de Certificação. Para sistemas de produção, é aconselhável não usar certificados autoassinados, mas em vez disso contar com certificados assinados por uma Autoridade de certificação.

- O parâmetro `label` especifica o nome para o certificado, que os interceptores consultarão para receber as informações necessárias.
- O parâmetro DN especifica os detalhes do **Nome Distinto** (DN), que deve ser exclusivo para cada usuário.

4. Agora criamos o banco de dados de chaves, é necessário configurar a propriedade dele e verificar se ele está ilegível para todos os outros usuários.

```
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. Repita as etapas 1-4 para o usuário bob

Resultados

Os dois usuários `alice` e bob agora possuem cada um certificado autoassinado.

4. Criando keystore.conf

Sobre esta tarefa

Deve-se apontar os interceptores do Advanced Message Security para o diretório no qual os bancos de dados de chave e certificados estão localizados. Isso é feito por meio do arquivo `keystore.conf`, que mantém essas informações em texto sem formatação. Cada usuário deve ter um arquivo `keystore.conf` separado na pasta `.mqs`. Esta etapa deve ser feita para ambos, `alice` e bob.

O conteúdo de `keystore.conf` deve ter o formato:

```
cms.keystore = <dir>/keystore_file
cms.certificate = certificate_label
```

Exemplo

Para este cenário, o conteúdo do `keystore.conf` será o seguinte:

```
cms.keystore = /home/alice/.mqs/alicekey
cms.certificate = Alice_Cert
```

Nota:

- O caminho para o arquivo keystore deve ser fornecido sem extensão de arquivo.
- Há os seguintes formatos de armazenamento de chaves: CMS (Cryptographic Message Syntax), JKS (Java Keystore) e JCEKS (Java Cryptographic Extension Keystore). Para obter informações adicionais, consulte “Estrutura do arquivo de configuração do keystore (keystore.conf)” na página 300.
- `HOME/.mqs/keystore.conf` é o local padrão no qual o Advanced Message Security procura o arquivo `keystore.conf`. Para obter informações sobre como usar um local não padrão para o `keystore.conf`, consulte “Usando keystores e certificados” na página 299.

5. Compartilhando os certificados

Sobre esta tarefa

Compartilhe os certificados entre os dois bancos de dados de chaves para que cada usuário possa identificar com sucesso o outro. Isso é feito extraindo cada certificado público do usuário para um arquivo, que é, então, incluído no banco de dados de chaves do outro usuário.

Nota: Tome cuidado para usar a opção *extract* e não a opção *export*. *Extract* obtém a chave pública do usuário, enquanto *export* obtém a chave pública e a privada. Usar *export* por engano comprometeria completamente o seu aplicativo, transmitindo a sua chave privada.

Procedimento

1. Extraia o certificado identificando `alice` para um arquivo externo:

```
runmqakm -cert -extract -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Alice_Cert
-target alice_public.arm
```

2. Inclua o certificado para o keystore bob's:

```
runmqakm -cert -add -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Alice_Cert -file
alice_public.arm
```

3. Repita a etapa para bob:

```
 runmqakm -cert -extract -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Bob_Cert -target
bob_public.arm
```

4. Inclua o certificado para bob para o keystore `alice`'s:

```
runmqakm -cert -add -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Bob_Cert -file
bob_public.arm
```

Resultados

Os dois usuários, `alice` e bob, agora podem identificar um ao outro com êxito como tendo criado e compartilhado certificados autoassinados.

Como proceder a seguir

Verifique se o certificado está no keystore executando os comandos a seguir que imprimem seus detalhes:

```
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. Definindo a política de filas

Sobre esta tarefa

Com o gerenciador de filas criado e os interceptores preparados para interceptar mensagens e acessar chaves de criptografia, podemos começar a definir políticas de proteção no `QM_VERIFY_AMS` usado o comando `setmqspl`. Consulte setmqspl para obter mais informações sobre este comando. Cada nome da política deve ser o mesmo que o nome da fila para a qual ela deve ser aplicada.

Exemplo

Este é um exemplo de uma política definida para a fila `TEST.Q`. Neste exemplo, as mensagens são assinadas pelo usuário `alice` usando o algoritmo SHA1 algoritmo e criptografados usando o algoritmo 256-bit AES.`alice` é o único emissor válido e bob é o único receptor das mensagens nesta fila:

```
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"
```

Nota: Os DNs correspondem exatamente aos especificados no respectivo certificado do usuário do banco de dados de chaves.

Como proceder a seguir

Para verificar a política que você definiu, emita o comando a seguir:

```
dspmqspl -m QM_VERIFY_AMS
```

Para imprimir os detalhes da política como um conjunto de comandos `setmqspl`, o sinalizador `-export`. Isso permite armazenar políticas já definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Testando a configuração

Sobre esta tarefa

Ao executar programas diferentes em diferentes usuários é possível verificar se o aplicativo foi configurado corretamente.

Procedimento

1. Mude para o diretório que contém as amostras. Se o MQ estiver instalado em um local não padrão, este poderá estar em um local diferente.

   ```
   cd /opt/mqm/samp/bin
   ```

2. Alterne o usuário para ser executado como `alice`

   ```
   su alice
   ```

3. Como o usuário `alice`, coloque uma mensagem usando um aplicativo de amostra:

   ```
   ./amqsput TEST.Q QM_VERIFY_AMS
   ```

4. Digite o texto da mensagem e em seguida pressione Enter.
5. Pare de executar como o usuário `alice`

   ```
   exit
   ```

6. Alterne o usuário para ser executado como bob

   ```
   su bob
   ```

7. Como o usuário bob, obtenha uma mensagem usando um aplicativo de amostra:

   ```
   ./amqsget TEST.Q QM_VERIFY_AMS
   ```

Resultados

Se o aplicativo foi configurado corretamente para ambos os usuários, a mensagem do usuário `alice`será exibida quando bob executar o aplicativo de obtenção.

8. Testando a criptografia

Sobre esta tarefa

Para verificar se a criptografia está ocorrendo conforme o esperado, crie uma fila de alias que faça referência à fila original `TEST.Q`. Esta fila de alias não terá uma política de segurança e, portanto, nenhum usuário terá as informações para decriptografar a mensagem e, portanto, os dados criptografados serão mostrados.

Procedimento

1. Usando o comando **runmqsc** no gerenciador de filas QM_VERIFY_AMS, crie uma fila de alias.

   ```
   DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
   ```

2. Conceda a bob acesso para procurar da fila de alias

   ```
   setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
   ```

3. Como o usuário `alice`, coloque outra mensagem usando um aplicativo de amostra como antes:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Como o usuário bob, procure a mensagem usando um aplicativo de amostra por meio da fila de alias dessa vez:

```
./amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. Como o usuário bob, obtenha a mensagem usando um aplicativo de amostra a partir da fila local:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Resultados

A saída do aplicativo `amqsbcg` mostrará os dados criptografados que estão na fila, provando que a mensagem foi criptografada.

Guia de iniciação rápida para clientes Java

Use este guia para configurar rapidamente IBM Advanced Message Security para fornecer segurança de mensagem para aplicativos Java que se conectam usando ligações de cliente. No momento em que você concluí-lo, você terá criado um armazenamento de chave para verificar as identidades do usuário e definido políticas de assinatura / criptografia para o seu gerenciador de filas.

Antes de começar

Assegure-se de ter os componentes apropriados instalados conforme descrito no **Guia de Iniciação Rápida** (Windows ou UNIX).

1. Criando um gerenciador de filas e uma fila

Sobre esta tarefa

Todos os exemplos a seguir usam uma fila nomeada `TEST.Q` para passar mensagens entre aplicativos. Advanced Message Security usa interceptores para assinar e criptografar mensagens no ponto em que elas entram na infraestrutura do WebSphere MQ por meio da interface padrão do WebSphere MQ A configuração básica é feita no WebSphere MQ e é configurada nas etapas a seguir.

Procedimento

1. Crie um gerenciador de filas

```
crtmqm QM_VERIFY_AMS
```

2. Iniciar o Gerenciador de Filas

```
strmqm QM_VERIFY_AMS
```

3. Crie e inicie um listener inserindo os comandos a seguir no **runmqsc** para o gerenciador de filas `QM_VERIFY_AMS`

```
DEFINE LISTENER(AMS.LSTR) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR)
START LISTENER(AMS.LSTR)
```

4. Crie um canal para os nossos aplicativos para se conectar inserindo o comando a seguir em **runmqsc** para o gerenciador de filas `QM_VERIFY_AMS`

```
DEFINE CHANNEL(AMS.SVRCONN) CHLTYPE(SVRCONN)
```

5. Crie uma fila chamada `TEST.Q` inserindo o comando a seguir em **runmqsc** para o gerenciador de filas `QM_VERIFY_AMS`

```
DEFINE QLOCAL(TEST.Q)
```

Resultados

Se o procedimento foi concluído com sucesso, o comando a seguir inserido em **runmqsc** irá exibir detalhes sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Criando e autorizando usuários

Sobre esta tarefa

Há dois usuários que aparecem em nosso cenário: alice, o remetente e bob, o destinatário Para usar a fila do aplicativo, esses usuários precisam receber autoridade para usá-la. Além disso, para usar com sucesso as políticas de proteção que vamos definir estes usuários devem ser concedidos acesso a algumas filas do sistema. Para obter mais informações sobre o comando **setmqaut** , consulte **setmqaut** .

Procedimento

1. Crie os dois usuários conforme descrito no **Guia de iniciação rápida** (Windows ou UNIX) para sua plataforma.
2. Autorize os usuários para se conectarem ao gerenciador de filas e para trabalhar com a fila

```
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
```

3. Também deve-se permitir que os dois usuários naveguem na fila de política do sistema e coloquem mensagens na fila de erros.

```
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
```

Resultados

Os usuários agora são criados e as autoridades necessárias concedidas a eles.

Como proceder a seguir

Para verificar se as etapas foram executadas corretamente, use as amostras JmsProducer e JmsConsumer , conforme descrito na seção “7. Testando a configuração” na página 293

3. Criando banco de dados de chaves e certificados

Sobre esta tarefa

Para criptografar a mensagem para o interceptor requer a chave pública dos usuários de envio. Portanto, o banco de dados de chaves das identidades do usuário mapeado para chaves pública e privada deve ser criado. No sistema real, no qual os usuários e aplicativos são dispersos por meio de vários computadores, cada usuário teria seu próprio keystore. Da mesma forma, nesse guia, criamos banco de dados de chaves para alice e bob e compartilhamos os certificados de usuário entre eles.

Nota: Neste guia, nós usamos aplicativos de amostra gravados em Java conectando usando ligações de cliente Se você planeja usar aplicativos Java usando ligações locais ou aplicativos C, deverá criar um keystore e certificados CMS usando o comando **runmqakm** . Isso é mostrado no **Guia de Início Rápido** (Windows ou UNIX).

Procedimento

1. Crie um diretório no qual criar seu keystore, por exemplo, /home/alice/.mqs. Você pode desejar criá-lo no mesmo diretório usado pelo **Guia de Iniciação Rápida** (Windows ou UNIX) para sua plataforma.

Nota: Esse diretório será referido como *keystore-dir* nas etapas a seguir

2. Crie um novo keystore e o certificado que identifica o usuário `alice` para uso na criptografia

   Nota: O comando **keytool** é parte do JRE.

   ```
   keytool -genkey -alias Alice_Java_Cert -keyalg RSA -keystore keystore-dir/keystore.jks
   -storepass passw0rd
   -dname "CN=alice, O=IBM, C=GB" -keypass passw0rd
   ```

   Nota:

   - Se o *keystore-dir* contiver espaços, deve-se colocar entre aspas ao redor o nome completo de seu keystore
   - É aconselhável usar uma senha forte para proteger o keystore.
   - Para o propósito desse guia, estamos usando certificado autoassinado que pode ser criado sem o uso de uma Autoridade de Certificação. Para sistemas de produção, é aconselhável não usar certificados autoassinados, mas, em vez disso, confiar em certificados assinados por uma autoridade de certificação.
   - O parâmetro `alias` especifica o nome para o certificado, que os interceptores consultarão para receber as informações necessárias.
   - O parâmetro dname especifica os detalhes do **Nome Distinto** (DN), que deve ser exclusivos para cada usuário.

3. No UNIX, assegure-se de que o keystore seja legível

   ```
   chmod +r keystore-dir/keystore.jks
   ```

4. Repita as etapas 1-4 para o usuário bob

Resultados

Os dois usuários `alice` e bob agora possuem cada um certificado autoassinado.

4. Criando keystore.conf

Sobre esta tarefa

Deve-se apontar os interceptores do Advanced Message Security para o diretório no qual os bancos de dados de chave e certificados estão localizados. Isso é feito por meio do arquivo `keystore.conf`, que contém essas informações no formato de texto simples. Cada usuário deve ter um arquivo `keystore.conf` separado. Esta etapa deve ser feita para ambos `alice` e bob.

Exemplo

Para esse cenário, o conteúdo do `keystore.conf` para `alice` será o seguinte:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Alice_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = passw0rd
JKS.key_pass = passw0rd
JKS.provider = IBMJCE
```

Para esse cenário, o conteúdo do `keystore.conf` para bob será o seguinte:

```
JKS.keystore = keystore-dir/keystore
JKS.certificate = Bob_Java_Cert
JKS.encrypted = no
JKS.keystore_pass = passw0rd
JKS.key_pass = passw0rd
JKS.provider = IBMJCE
```

Nota:

- O caminho para o arquivo keystore deve ser fornecido sem extensão de arquivo.
- Se você já tiver um `keystore.conf` porque seguiu o **Guia de iniciação rápida** (Windows ou UNIX), poderá editar o existente para incluir nas linhas acima.
- Para obter informações adicionais, consulte "Estrutura do arquivo de configuração do keystore (keystore.conf)" na página 300.

5. Compartilhando os certificados

Sobre esta tarefa

Compartilhe os certificados entre os dois keystores para que cada usuário pode identificar com sucesso o outro. Isso é feito extraindo cada certificado do usuário e importando-o para o keystore do outro usuário.

Nota: Os termos *extract* e *export* são usados de forma diferente por ferramentas de certificado diferentes. Por exemplo, a ferramenta do IBM GSKit Keyman (ikeyman) faz uma distinção que você *extrai* certificados (chaves públicas) e *exporta* chaves privadas. Essa distinção é extremamente importante para ferramentas que oferecem ambas as opções, pois usar *export* por engano comprometeria completamente o seu aplicativo, transmitindo a sua chave privada. Como a distinção é tão importante, a documentação do WebSphere MQ esforça-se para usar esses termos de forma consistente. No entanto, a keytool do Java fornece uma opção da linha de comandos denominada *exportcert* que extrai apenas a chave pública. Por esses motivos, o procedimento a seguir refere-se a *extrair* certificados usando a opção *exportcert*.

Procedimento

1. Extraia o certificado que identifica `alice`.

   ```
   keytool -exportcert -keystore alice-keystore-dir/keystore.jks -storepass passw0rd
   -alias Alice_Java_Cert -file alice-keystore-dir/Alice_Java_Cert.cer
   ```

2. Importe o certificado identificando `alice` no keystore que bob usará. Quando solicitado indique que você irá confiar nesse certificado.

   ```
   keytool -importcert -file alice-keystore-dir/Alice_Java_Cert.cer -alias Alice_Java_Cert
   -keystore bob-keystore-dir/keystore.jks -storepass passw0rd
   ```

3. Repita as etapas para bob

Resultados

Os dois usuários, `alice` e bob, agora podem identificar um ao outro com êxito como tendo criado e compartilhado certificados autoassinados.

Como proceder a seguir

Verifique se o certificado está no keystore executando os comandos a seguir que imprimem seus detalhes:

```
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. Definindo a política de filas

Sobre esta tarefa

Com o gerenciador de filas criado e os interceptores preparados para interceptar mensagens e acessar chaves de criptografia, podemos começar a definir políticas de proteção no `QM_VERIFY_AMS` usado o comando `setmqspl`. Consulte setmqspl para obter mais informações sobre este comando. Cada nome da política deve ser o mesmo que o nome da fila para a qual ela deve ser aplicada.

Exemplo

Este é um exemplo de uma política definida na fila TEST.Q, assinada pelo usuário alice usando o algoritmo SHA1 e criptografada usando o algoritmo 256-bit AES para o usuário 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"
```

Nota: Os DNs correspondem exatamente aos especificados no respectivo certificado do usuário do banco de dados de chaves.

Como proceder a seguir

Para verificar a política que você definiu, emita o comando a seguir:

```
dspmqspl -m QM_VERIFY_AMS
```

Para imprimir os detalhes da política como um conjunto de comandos setmqspl, o sinalizador -export. Isso permite armazenar políticas já definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Testando a configuração

Antes de começar

Assegure-se de que a versão do Java que você está usando possui os arquivos de políticas JCE sem restrição instalados.

Nota: A versão do Java fornecida na instalação do WebSphere MQ já possui esses arquivos de políticas. Ela pode ser localizada em *MQ_INSTALLATION_PATH*/java/bin.

Sobre esta tarefa

Ao executar programas diferentes em diferentes usuários é possível verificar se o aplicativo foi configurado corretamente. Consulte o **Guia de Iniciação Rápida** (Windows ou UNIX) para sua plataforma, para obter detalhes sobre como executar programas em diferentes usuários.

Procedimento

1. Para executar esses aplicativos de amostra JMS, use a configuração CLASSPATH para sua plataforma, conforme mostrado em Variáveis de ambiente usadas por IBM WebSphere MQ classes para JMS para assegurar que o diretório de amostras seja incluído
2. Como o usuário alice, coloque uma mensagem usando um aplicativo de amostra, se conectando como um cliente:

   ```
   java JMSProducer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

3. Como o usuário bob, obtenha uma mensagem usando um aplicativo de amostra, se conectando como um cliente:

   ```
   java JMSConsumer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

Resultados

Se o aplicativo foi configurado corretamente para ambos os usuários, a mensagem do usuário aliceserá exibida quando bob executar o aplicativo de obtenção.

Protegendo filas remotas

Para proteger totalmente as conexões de fila remota, a mesma política deve ser configurada na fila remota e na fila local para a qual as mensagens serão transmitidas.

Quando uma mensagem é colocada em uma fila remota, o Advanced Message Security intercepta a operação e processa a mensagem de acordo com a política configurada para a fila remota. Por exemplo, para uma política de criptografia a mensagem é criptografada antes de ser transmitida para que o WebSphere MQ a manuseie. Depois que Advanced Message Security tiver processado a mensagem colocada em uma fila remota, o WebSphere MQ a colocará em uma fila de transmissão associada e a redirecionará para o gerenciador de filas de destino e para a fila de destino.

Quando uma operação GET for executada na fila local, o Advanced Message Security tentará decodificar a mensagem de acordo com a política configurada na fila local. Para que a operação seja bem-sucedida, a política usada para decriptografar a mensagem deve ser idêntica à que foi usada para criptografá-la. Qualquer discrepância fará com que a mensagem seja rejeitada.

Se, por algum motivo ambas as políticas não puderem ser configuradas ao mesmo tempo, um suporte ao lançamento em estágios será fornecido. A política pode ser configurada em uma fila local com o sinalizador de tolerância ligado, o que indica que uma política associada a uma fila pode ser ignorada quando uma tentativa de recuperar uma mensagem da fila envolve uma mensagem que não possui o conjunto de política de segurança. Neste caso, GET tentará decriptografar a mensagem, mas permitirá que as mensagens não criptografadas sejam entregues. Dessa maneira as políticas em filas remotas podem ser configuradas após a filas locais terem sido protegidas (e testadas).

Não se esqueça: Remova o sinalizador de tolerância quando a apresentação do Advanced Message Security for concluída.

Referências relacionadas

setmqspl (configurar política de segurança)

Roteando mensagens protegidas usando o Message Broker do WebSphere

IBM Advanced Message Security pode proteger mensagens em uma infraestrutura em que o WebSphere Message Broker versão 8.0.0.1 (ou posterior) está instalado. Você deve entender a natureza de ambos os produtos antes de aplicar a segurança no ambiente do Message Broker do WebSphere

Sobre esta tarefa

O Advanced Message Security fornece segurança de ponta a ponta da carga útil da mensagem. Isso significa que apenas as partes especificadas como emissores e destinatários válidos de uma mensagem são capazes de produzir ou recebê-la. Isso significa que, para proteger as mensagens que fluem pelo WebSphere Message Broker, é possível permitir que o WebSphere Message Broker processe mensagens sem conhecer seu conteúdo (Cenário 1) ou torná-lo um usuário autorizado capaz de receber e enviar mensagens (Cenário 2).

Cenário 1-O Message Broker não pode ver o conteúdo da mensagem

Antes de começar

Você deve ter seu WebSphere Message Broker conectado a um gerenciador de filas existente. Substitua *QMgrName* com esse nome do gerenciador de filas existente nos comandos a seguir.

Sobre esta tarefa

Neste cenário, Alice coloca uma mensagem protegida em uma fila de entrada `QIN`. Com base na propriedade de mensagem `routeTo`, a mensagem é roteada para *bob* (`QBOB`),[1](`QCECIL`) ou a fila padrão (`QDEF`). O roteamento é possível porque o Advanced Message Security protege apenas a carga útil da mensagem e não seus cabeçalhos e propriedades que permanecem desprotegidos e podem ser lidos pelo WebSphere Message Broker. Advanced Message Security é usado apenas por *alice*, *bob* e *cecil* Não é necessário instalá-lo ou configurá-lo para o Message Broker do WebSphere

WebSphere O Message Broker recebe a mensagem protegida da fila de alias desprotegido para evitar qualquer tentativa de decriptografar a mensagem Se fosse usar a fila protegida diretamente a mensagem seria colocada na fila DEAD LETTER como impossível de decriptografar. A mensagem é roteada pelo WebSphere Message Broker e chega na fila de destino inalterada. Portanto, ele ainda é assinado pelo

[1] cecil's

autor original (ambos *bob* e *cecil* aceitam apenas mensagens enviadas por *alice*) e protegido como antes (apenas *bob* e *cecil* podem lê-lo)... O Message Broker do WebSphere coloca a mensagem roteada em um alias desprotegido Os destinatários recuperam a mensagem a partir de uma fila de saída protegida em que IBM WebSphere MQ AMS decriptografará a mensagem de modo transparente.

Procedimento

1. Configure *alice*, *bob* e *cecil* para usar o Advanced Message Security conforme descrito no **Guia de iniciação rápida** (Windows ou UNIX).

   Assegure-se de que as etapas a seguir sejam concluídas:

   - Criando e autorizando usuários
   - Criando banco de dados de chaves e certificados
   - Criando keystore.conf
2. Forneça o certificado da *alice* para *bob* e *cecil* para que *alice* possa ser identificada por eles ao verificarem assinaturas digitais em mensagens.

   Faça isso extraindo o certificado que identifica *alice* em um arquivo externo e, em seguida, incluindo o certificado extraído nos keystores *do bob* e *da cecil*. É importante que você use o método descrito na **Tarefa 5. Certificados de compartilhamento** no **Guia de iniciação rápida** (Windows ou UNIX).
3. Forneça os certificados de *bob* e *do cecil* para *alice*, para que *alice* possa enviar mensagens criptografadas para *bob* e *cecil*.

   Faça isso usando o método especificado na etapa anterior.
4. Em seu gerenciador de filas, defina as filas locais chamadas `QIN`, `QBOB`, `QCECIL` e `QDEF`.

   ```
   DEFINE QLOCAL(QIN)
   ```
5. Configure a política de segurança para a fila `QIN` para uma configuração elegível. Use a configuração idêntica para as filas `QBOB`, `QCECIL` e `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"
   ```

   Este cenário supõe a política de segurança na qual *alice* é o único emissor autorizado e *bob* e *cecil* são os destinatários.
6. Defina as filas de alias `AIN`, `ABOB` e `ACECIL` fazendo referência à filas locais `QIN`, `QBOB` e `QCECIL` respectivamente.

   ```
   DEFINE QALIAS(AIN) TARGET(QIN)
   ```
7. Verifique se a configuração de segurança para o alias especificado na etapa anterior não está presente; caso contrário, defina sua política para NONE.

   ```
   dspmqspl -m QMgrName -p AIN
   ```
8. No WebSphere Message Broker, crie um fluxo de mensagens para rotear as mensagens que chegam na fila de alias `AIN` para o nó BOB, CECIL ou DEF, dependendo da propriedade `routeTo` da mensagem. Para fazê-lo:
   a) Crie um nó `MQInput` chamado `IN` e designe o alias `AIN` como seu nome da fila.
   b) Crie `MQOutput` nós chamados BOB, `CECIL` e DEF e designe filas de alias `ABOB`, `ACECIL` e `ADEF` como seus respectivos nomes de filas..
   c) Crie um nó de rota e chame-o TEST.
   d) Conecte o nó `IN` ao terminal de entrada do nó TEST.
   e) Crie os terminais de saída bob e `cecil` para o nó TEST.
   f) Conecte o terminal de saída bob ao nó BOB.
   g) Conecte o terminal de saída `cecil` ao nó `CECIL`.

h) Conecte o nó DEF para o terminal de saída padrão.

i) Aplique as regras a seguir:

```
$Root/MQRFH2/usr/routeTo/text()="bob"
$Root/MQRFH2/usr/routeTo/text()="cecil"
```

9. Implemente o fluxo de mensagens no componente de tempo de execução do WebSphere Message Broker.
10. Executando como o usuário `Alice`, coloque uma mensagem que também contém uma propriedade de mensagem chamada `routeTo` com um valor de bob ou `cecil`. Executar o aplicativo de amostra **amqsstm** permitirá que você faça isso.

```
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. Executando como o usuário *bob*, recupere a mensagem da fila `QBOB` usando o aplicativo de amostra **amqsget**.

Resultados

Quando *alice* colocar uma mensagem na fila `QIN` a mensagem será protegida. Ele é recuperado no formulário protegido pelo Message Broker do WebSphere da fila de alias `AIN` . WebSphere O Message Broker decide para onde rotear a mensagem que lê a propriedade `routeTo` que é, como todas as propriedades, não criptografada. WebSphere O Message Broker coloca a mensagem no alias desprotegido apropriado, evitando sua proteção adicional. Quando recebida por *bob* ou *cecil* da fila, a mensagem será decriptografada e a assinatura digital será verificada.

Cenário 2-O Message Broker pode ver o conteúdo da mensagem

Sobre esta tarefa

Neste cenário, um grupo de indivíduos tem permissão para enviar mensagens para o Message Broker do WebSphere . Outro grupo está autorizado a receber mensagens que são criadas pelo Message Broker do WebSphere . A transmissão entre as partes e o Message Broker do WebSphere não pode ser espionada.

Lembre-se de que o WebSphere Message Broker lê políticas de proteção e certificados somente quando uma fila é aberta, portanto, você deve recarregar o grupo de execução após fazer quaisquer atualizações nas políticas de proteção para que as mudanças entrem em vigor.

```
mqsireload execution-group-name
```

Se o Message Broker do WebSphere for considerado uma parte autorizada com permissão para ler ou assinar a carga útil da mensagem, deve-se configurar o Advanced Message Security para o usuário que inicia o serviço do Message Broker do WebSphere . Esteja ciente de que não é necessariamente o mesmo usuário que coloca / obtém as mensagens nas filas nem o usuário que cria e implementa os aplicativos Message Broker do WebSphere .

Procedimento

1. Configure *alice*, *bob*, *cecil* e *dave* e o usuário do serviço WebSphere Message Broker para usar o Advanced Message Security conforme descrito no **Guia de Iniciação Rápida** (Windows ou UNIX).

   Assegure-se de que as etapas a seguir sejam concluídas:

   - Criando e autorizando usuários

- Criando banco de dados de chaves e certificados
- Criando keystore.conf

2. Forneça os certificados *alice*, *bob*, *cecil* e *dave's* para o usuário do serviço do WebSphere Message Broker.

   Faça isso extraindo para arquivos externos cada um dos certificados que identificam *alice*, *bob*, *cecil* e *dave*, em seguida, incluindo os certificados extraídos no keystore do WebSphere Message Broker. É importante que você use o método descrito na **Tarefa 5. Certificados de compartilhamento** no **Guia de iniciação rápida** (Windows ou UNIX).
3. Forneça o certificado do usuário do serviço do WebSphere Message Broker para *alice*, *bob*, *cecil* e *dave*

   Faça isso usando o método especificado na etapa anterior.

   Nota: *Alice* e *bob* precisam do certificado do usuário do Message Broker do WebSphere para criptografar as mensagens corretamente. O usuário do serviço do Message Broker do WebSphere precisa de certificados *alice's* e *bob's* para verificar os autores das mensagens O usuário do serviço do Message Broker do WebSphere precisa de certificados do *cecil* e do *dave* para criptografar as mensagens para eles. *cecil* e *dave* precisam do certificado do usuário do Message Broker do WebSphere para verificar se a mensagem é proveniente do Message Broker do WebSphere .
4. Defina uma fila local denominada `IN` e defina a política de segurança com *alice* e *bob* especificados como autores e o usuário do serviço do WebSphere Message Broker especificado como destinatário:

   ```
   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. Defina uma fila local chamada OUT e defina a política de segurança com o usuário do serviço do WebSphere Message Broker especificado como autor e *cecil* e *dave* especificado como destinatários:

   ```
   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. No WebSphere , o Message Broker cria um fluxo de mensagens com um nó `MQInput` e `MQOutput` Configure o nó `MQInput` para usar a fila `IN` e o nó `MQOutput` para usar a fila `OUT`.
7. Implemente o fluxo de mensagens no componente de tempo de execução do WebSphere Message Broker.
8. Executando como os usuários *alice* ou *bob*, coloque uma mensagem na fila `IN` usando o aplicativo de amostra **amqsput**.
9. Executando como os usuários *cecil* ou *dave*, recupere a mensagem da fila OUT usando o aplicativo de amostra **amqsget**.

Resultados

As mensagens enviadas por *alice* ou *bob* para a fila de entrada `IN` são criptografadas permitindo que apenas o WebSphere Message Broker as leia. WebSphere O Message Broker aceitará apenas mensagens de *alice* e *bob* e rejeitará quaisquer outras. As mensagens aceitas serão processadas apropriadamente e, em seguida, assinadas e criptografadas com as chaves *cecil's* e *dave's* antes de serem colocadas na fila de saída OUT Apenas *cecil* e *dave* são capazes de lê-lo, as mensagens não assinadas pelo WebSphere Message Broker são rejeitadas

Usando o IBM WebSphere MQ Advanced Message Security com o IBM WebSphere MQ Managed File Transfer

Este cenário explica como configurar o Advanced Message Security para fornecer privacidade de mensagem para dados que estão sendo enviados por meio de um IBM WebSphere MQ Managed File Transfer.

Antes de começar

Assegure-se de que você tenha o componente do Advanced Message Security instalado na instalação do WebSphere MQ hospedando as filas usadas pelo IBM WebSphere MQ Managed File Transfer que você deseja proteger.

Se os agentes do IBM WebSphere MQ Managed File Transfer estiverem se conectando no modo de ligações, assegure-se de ter o componente GSKit instalado em sua instalação local.

Sobre esta tarefa

Quando a transferência de dados entre dois agentes do IBM WebSphere MQ Managed File Transfer for interrompida, dados possivelmente confidenciais poderão permanecer desprotegido nas filas do WebSphere MQ subjacentes usadas para gerenciar a transferência. Este cenário explica como configurar e usar o Advanced Message Security para proteger tais dados no IBM WebSphere MQ Managed File Transfer.

Neste cenário, consideramos uma topologia simples que consiste em uma máquina com duas IBM WebSphere MQ Managed File Transfer filas e dois agentes, AGENT1 e AGENT2, compartilhando um único gerenciador de filas, hubQM, conforme descrito no cenário Transferência de arquivos básica usando os scripts Ambos os agentes se conectam da mesma maneira, no modo de ligações ou no modo cliente.

1. Criando certificados

Antes de começar

Esse cenário usa um modelo simples em que um usuário `ftagent` em um grupo FTAGENTS é usado para executar os processos do agente IBM WebSphere MQ Managed File Transfer. Se você estiver usando os seus próprios nomes do usuário e do grupo, mude os comandos de modo correspondente.

Sobre esta tarefa

O Advanced Message Security usa a criptografia de chave pública para assinar e/ou criptografar mensagens em filas protegidas.

Nota:

- Se os agentes do IBM WebSphere MQ Managed File Transfer estiverem em execução no modo de ligação, os comandos usados para criar um keystore CMS (Cryptographic Message Syntax) serão detalhados no **Guia de Iniciação Rápida** (Windows ou UNIX) para sua plataforma.
- Se seus agentes do IBM WebSphere MQ Managed File Transfer estiverem em execução no modo cliente, os comandos necessários para criar um JKS (Java Keystore) serão detalhados no “Guia de iniciação rápida para clientes Java” na página 289.

Procedimento

1. Crie um certificado autoassinado para identificar o usuário `ftagent` conforme detalhado no Guia de iniciação rápida apropriado.
   Use um Nome distinto (DN) conforme a seguir:

   ```
   CN=ftagent, OU=MFT, O=IBM, L=Hursley, ST=Hampshire, C=GB
   ```

2. Crie um arquivo `keystore.conf` para identificar o local do keystore e o certificado dentro dele conforme detalhado no Guia de iniciação rápida apropriado.

2. Configurando a proteção de mensagens

Sobre esta tarefa

É necessário definir uma política de segurança para a fila de dados usada pelo AGENT2 usando o comando **`setmqspl`**. Nesse cenário, o mesmo usuário é usado para iniciar ambos os agentes, e, portanto, o signatário e o destinatário DN são iguais e corresponderem ao certificado gerado.

Procedimento

1. Encerre os agentes do IBM WebSphere MQ Managed File Transfer em preparação para a proteção usando o comando **`fteStopAgent`**.
2. Crie uma política de segurança para proteger a fila SYSTEM.FTE.DATA.AGENT2.

```
setmqspl -m hubQM -p SYSTEM.FTE.DATA.AGENT2 -s SHA1 -a "CN=ftagent, OU=MFT, O=IBM,
L=Hursley, ST=Hampshire, C=GB"
-e AES128 -r "CN=ftagent, OU=MFT, O=IBM, L=Hursley, ST=Hampshire, C=GB"
```

3. Assegure que o usuário que executa o processo do agente do IBM WebSphere MQ Managed File Transfer tem acesso para navegar pela fila de política do sistema e colocar mensagens na fila de erros.

```
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. Reinicie os agentes do IBM WebSphere MQ Managed File Transfer usando o comando **fteStartAgent**.
5. Confirme se seus agentes foram reiniciados com sucesso usando o comando **fteListAgents** e verifique se os agentes estão em status READY.

Resultados

Você agora é capaz de enviar transferências do AGENT1 para AGENT2 e o conteúdo do arquivo será transmitido de forma segura entre os dois agentes.

Instalando o IBM WebSphere MQ Advanced Message Security

Instale o componente IBM WebSphere MQ Advanced Message Security em várias plataformas.

Sobre esta tarefa

Para obter os procedimentos de instalação completos, consulte Instalando o IBM WebSphere MQ Advanced Message Security .

Tarefas relacionadas

Desinstalando oIBM WebSphere MQ Advanced Message Security

Usando keystores e certificados

Para fornecer proteção criptográfica transparente para aplicativos WebSphere MQ, o Advanced Message Security usa o arquivo keystore, em que certificados de chave pública e uma chave privada são armazenados.

No Advanced Message Security, usuários e aplicativos são representados por identidades de infraestrutura de chave pública (PKI). Esse tipo de identidade é usado para assinar e criptografar mensagens. A identidade PKI é representada pelo campo **nome distinto (DN)** do sujeito em um certificado que está associado com mensagens assinadas e criptografadas. Para um usuário ou aplicativo criptografar suas mensagens eles requerem acesso ao arquivo keystore no qual os certificados e chaves privadas e públicas associadas são armazenados.

O local do keystore é fornecido no arquivo de configuração do keystore, que é `keystore.conf` por padrão. Cada usuário do Advanced Message Security deve ter o arquivo de configuração keystore que aponta para um arquivo keystore. Advanced Message Security aceita o formato de arquivos keystore a seguir: `.kdb`, `.jceks`, `.jks`.

O local padrão do arquivo `keystore.conf` é:

- Em plataformas UNIX : `$HOME/.mqs/keystore.conf`
- Em plataformas Windows: *%HOMEDRIVE%%HOMEPATH%*`\.mqs\keystore.conf`

Se você estiver usando um nome do arquivo keystore e local especificados, é necessário usar os comandos a seguir

- Para Java: `java -D` *`MQS_KEYSTORE_CONF=path/filename app_name`*
- Para C Client e Servidor:
  - No UNIX and Linux: `export MQS_KEYSTORE_CONF=`*`path/filename`*
  - No Windows: `set MQS_KEYSTORE_CONF=`*`path\filename`*

Nota: O caminho em Windows pode e deve especificar a letra da unidade se mais de uma letra de unidade estiver disponível.

Conceitos relacionados

"Nomes distintos do emissor" na página 313
Os nomes distintos (DNs) do emissor identificam usuários que estão autorizados a colocar mensagens em uma fila.

"Nomes distintos do destinatário" na página 314
Os nomes distintos (DN) do destinatário identificam usuários que estão autorizados a recuperar mensagens em uma fila.

Estrutura do arquivo de configuração do keystore (keystore.conf)

O arquivo de configuração do keystore (`keystore.conf`) aponta Advanced Message Security para o local do keystore apropriado.

Há dois tipos de configuração de CMS e Java (JKS e JCEKS) As entradas de configuração do CMS são prefixadas com `cms.` e Java são prefixadas com `jks.` ou `jceks.`, dependendo do tipo de um keystore.

O arquivo de configuração, dependendo do tipo de arquivo de configuração, pode ter uma das seguintes estruturas:

```
cms.keystore = /<dir>/<keystore_file>
cms.certificate = certificate_label

jceks.keystore = <dir>/Keystore
jceks.certificate = <certificate_label>
jceks.encrypted = no
jceks.keystore_pass = <password>
jceks.key_pass = <password>
jceks.provider = IBMJCE

jks.keystore = <dir>/Keystore
jks.certificate = <certificate_label>
jks.encrypted = no
jks.keystore_pass = <password>
jks.key_pass = <password>
jks.provider = IBMJCE
```

Os parâmetros de arquivo de configuração são definidos como seguir:

keystore
Caminho para o arquivo keystore.

Importante:

- O caminho para o arquivo keystore não deve incluir a extensão do arquivo.
- Para arquivos Java keystore, o IBM WebSphere MQ AMS suporta os seguintes formatos de arquivo: `.jks`, `.jceks`, `.jck`.

certificate
Etiqueta do certificado.

encrypted
Status da senha.

keystore_pass
Senha do arquivo de armazenamento de chaves.

Nota:

- Para o keystore CMS, o IBM WebSphere MQ AMS depende dos arquivos stash (`.sth`), enquanto que JKS e JCEKS podem requerer uma senha para o certificado e a chave privada do usuário.
- Armazenando senhas em texto simples é um risco para a segurança.

key_pass
Senha para chave privada do usuário.

Importante: Armazenar senhas em texto simples pode ser um risco para a segurança.

provider
O provedor de segurança Java que implementa algoritmos criptográficos requeridos pelo certificado keystore.

Nota: Atualmente, o IBMJCE é o único provedor suportado pelo Advanced Message Security.

Importante: As informações armazenadas no keystore são cruciais para o fluxo de dados enviados segura usando o WebSphere MQ, que é o motivo pelo qual os administradores de segurança devem prestar atenção especial ao designar permissões de arquivo para esses arquivos.

Aqui está um exemplo do arquivo `keystore.conf`:

```
cms.keystore = c:\Documents and Settings\Alice\AliceKeystore
cms.certificate = AliceCert

jceks.keystore = c:/Documents and Settings/Alice/AliceKeystore
jceks.certificate = AliceCert
jceks.encrypted = no
jceks.keystore_pass = <password>
jceks.key_pass = <password>
jceks.provider = IBMJCE
```

Tarefas relacionadas

“Protegendo as senhas no Java” na página 311
Armazenamento de keystore e senhas de chave privada como texto simples representam um risco de segurança, então o Advanced Message Security fornece uma ferramenta que pode misturar essas senhas usando a chave do usuário, que está disponível no arquivo keystore.

Intercepção de Agente do Canal de Mensagens (MCA)

A intercepção do MCA permite que um gerenciador de filas em execução sob IBM WebSphere MQ ative seletivamente as políticas a serem aplicadas para os canais de conexão do servidor.

A intercepção de MCA permite aos clientes que permanecem fora do IBM WebSphere MQ AMS continuem conectados a um gerenciador de filas e suas mensagens sejam criptografadas e decriptografadas.

A intercepção de MCA destina-se a fornecer a capacidade do IBM WebSphere MQ AMS quando o IBM WebSphere MQ AMS não puder ser ativado no cliente. Observe que usar a intercepção de MCA e um cliente ativado por IBM WebSphere MQ AMS leva à dupla proteção de mensagens que pode ser problemática para aplicativos de recebimento.

Se um erro `2085 (MQRC_UNKNOWN_OBJECT_NAME)` for relatado se você estiver usando um Version 7.5 ou cliente posterior para se conectar a um gerenciador de filas de uma versão anterior do produto, será necessário desativar IBM WebSphere MQ Advanced Message Security no cliente. Para obter mais informações, consulte “Desativando IBM WebSphere MQ Advanced Message Security no cliente” na página 304.

Arquivo de configuração do keystore

Por padrão, o arquivo de configuração keystore para intercepção de MCA é `keystore.conf` e está localizado no diretório `.mqs` no caminho do diretório HOME do usuário que iniciou o gerenciador de filas ou o listener. O keystore também pode ser configurado usando a variável de ambiente MQS_KEYSTORE_CONF. Para obter mais informações sobre como configurar o keystore do IBM WebSphere MQ AMS, veja “Usando keystores e certificados” na página 299.

Para ativar a intercepção MCA, deve-se fornecer o nome de um canal que você deseja usar no arquivo de configuração keystore. Para a intercepção de MCA, somente um tipo de keystore cms pode ser usado.

Para obter um exemplo de como configurar a intercepção de MCA, consulte “Exemplo de intercepção de MCA do IBM WebSphere MQ AMS” na página 302

Atenção: Deve-se concluir a autenticação de cliente e a criptografia nos canais selecionados, por exemplo, usando SSL e SSLPEER ou CHLAUTH TYPE(SSLPEERMAP), para assegurar que somente os clientes autorizados possam se conectar e usar esse recurso.

Exemplo de intercepção de MCA do IBM WebSphere MQ AMS

Uma tarefa de exemplo de como configurar uma intercepção de MCA do IBM WebSphere MQ AMS.

Antes de começar

Atenção: Deve-se concluir a autenticação de cliente e a criptografia nos canais selecionados, por exemplo, usando SSL e SSLPEER ou CHLAUTH TYPE(SSLPEERMAP), para assegurar que somente os clientes autorizados possam se conectar e usar esse recurso.

Sobre esta tarefa

Esta tarefa leva você através do processo de configuração do seu sistema para usar a intercepção de MCA e, em seguida, verificar a configuração.

Nota: Antes do IBM WebSphere MQ Version 7.5, o IBM WebSphere MQ AMS era um produto complementar que precisava ser instalado separadamente e ter interceptores configurados para proteger aplicativos. No Version 7.5 em diante, os interceptores são automaticamente incluídos e dinamicamente ativados no cliente do MQ e em ambientes de tempo de execução do servidor. Neste exemplo de intercepção de MCA, os interceptores são fornecidos no término do servidor do canal e um tempo de execução de cliente mais antigo é usado (na Etapa 12) para colocar uma mensagem desprotegida ao longo do canal para que ela possa ser vista para ser protegida pelos interceptores de MCA. Se este exemplo usasse um cliente do Version 7.5 ou mais recente, ele faria a mensagem ser protegida duas vezes, porque o interceptor de tempo de execução de cliente do MQ e o interceptor de MCA ambos protegeriam a mensagem assim que ela chegasse no MQ.

Atenção: Substitua `userID` no código com seu ID do usuário.

Procedimento

1. Crie o banco de dados de chave e certificados usando os seguintes a seguir para criar um shell script.

   Além disso, mude o **INSTLOC** e **KEYSTORELOC** ou executar os comandos necessários. Observe que pode não ser necessário para o certificado para o bob.

```
INSTLOC=/opt/mq75
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. Compartilhe os certificados entre os dois bancos de dados de chaves para que cada usuário possa identificar com sucesso o outro.

   É importante que você use o método descrito na **Tarefa 5. Certificados de compartilhamento** no **Guia de iniciação rápida** (Windows ou UNIX).

3. Crie `keystore.conf` com a configuração a seguir: `Keystore.conf location: /home/userID/ssl/ams1/`

```
cms.keystore = /home/userID/ssl/ams1/alicekey
cms.certificate.channel.SYSTEM.DEF.SVRCONN = alice_cert
```

4. Crie e inicie o gerenciador de filas AMSQMGR1
5. Defina um listener com *port* 14567 e *control* QMGR
6. Desative a autoridade do canal ou configure as regras para a autoridade do canal.
   Consulte SET CHLAUTH para obter mais informações.
7. Parar o gerenciador de fila.
8. Configure o keystore:

```
export MQS_KEYSTORE_CONF=/home/userID/ssl/ams1/keystore.conf
```

9. Inicie o gerenciador de filas no mesmo shell.
10. Configure a política de segurança e verifique:

```
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
```

    Consulte setmqspl e dspmqspl para obter mais informações.
11. Configure a configuração do canal:

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/127.0.0.1(14567)'
```

12. Execute **amqsputc** por meio de um cliente do MQ que não ativa automaticamente um interceptor de MCA; por exemplo um cliente do IBM WebSphere MQ Version 7.1 ou anterior. Coloque as duas mensagens a seguir:

```
/opt/mqm/samp/bin/amqsputc TESTQ TESTQMGR
```

13. Remova a política de segurança e verifique o resultado:

```
setmqspl -m AMSQMGR1 -p TESTQ -remove
dspmqspl -m AMSQMGR1
```

14. Procure a fila por meio de sua instalação do IBM WebSphere MQ Version 7.5:

```
/opt/mq75/samp/bin/amqsbcg TESTQ AMSQMGR1
```

    A saída de procura mostra as mensagens no formato criptografado.
15. Configure a política de segurança e verifique o resultado:

```
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. Execute **amqsgetc** por meio de sua instalação do IBM WebSphere MQ Version 7.5:

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

Tarefas relacionadas

“Guia de iniciação rápida para clientes Java” na página 289
Use este guia para configurar rapidamente IBM Advanced Message Security para fornecer segurança de mensagem para aplicativos Java que se conectam usando ligações de cliente. No momento em que você concluí-lo, você terá criado um armazenamento de chave para verificar as identidades do usuário e definido políticas de assinatura / criptografia para o seu gerenciador de filas.

Referências relacionadas

“Limitações conhecidas” na página 277
Aprenda sobre limitações do IBM WebSphere MQ Advanced Message Security.

Desativando IBM WebSphere MQ Advanced Message Security no cliente

Será necessário desativar IBM WebSphere MQ Advanced Message Security (AMS) no cliente se você estiver usando um cliente Version 7.5 ou posterior para se conectar a um gerenciador de filas de uma versão anterior do produto e um erro 2085 (MQRC_UNKNOWN_OBJECT_NAME) for relatado.

Sobre esta tarefa

A partir de Version 7.5, o IBM WebSphere MQ Advanced Message Security (AMS) é ativado automaticamente em um cliente IBM WebSphere MQ , portanto, por padrão, o cliente tenta verificar as políticas de segurança para objetos no gerenciador de filas. No entanto, os servidores em versões anteriores do produto, por exemplo Version 7.1, não têm o AMS ativado, o que faz com que um erro 2085 (MQRC_UNKNOWN_OBJECT_NAME) seja relatado

Se esse erro for relatado quando você estiver tentando se conectar a um gerenciador de filas a partir de uma versão anterior do produto, será possível desativar o AMS no cliente, conforme a seguir:

- Para clientes Java, de qualquer uma das maneiras a seguir:
  - V 7.5.0.4 Configurando uma variável de ambiente AMQ_DISABLE_CLIENT_AMS.
  - V 7.5.0.4 Configurando a propriedade do sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS.
  - V 7.5.0.5 Usando a propriedade DisableClientAMS, na sub-rotina **Security** no arquivo mqclient.ini.
- Para clientes C, de uma das maneiras a seguir:
  - V 7.5.0.4 Configurando uma variável de ambiente AMQ_DISABLE_CLIENT_AMS.
  - V 7.5.0.5 Usando a propriedade DisableClientAMS, na sub-rotina **Security** no arquivo mqclient.ini.

Procedimento

- Para desativar o AMS no cliente, use uma das opções a seguir:

  V 7.5.0.4 **Variável de ambiente AMQ_DISABLE_CLIENT_AMS**

  É necessário configurar essa variável nos seguintes casos:

  - Se você estiver usando o Java Runtime Environment (JRE) diferente do IBM Java Runtime Environment (JRE)
  - Se estiver usando o cliente Version 7.5ou posterior, IBM WebSphere MQ classes for Java ou IBM WebSphere MQ classes for JMS .

  Também é possível usar AMQ_DISABLE_CLIENT_AMS para desativar a funcionalidade do AMS para clientes C.

  Crie a variável de ambiente AMQ_DISABLE_CLIENT_AMS e configure-a como TRUE no ambiente em que o aplicativo está em execução. Por exemplo:

  ```
  export AMQ_DISABLE_CLIENT_AMS=TRUE
  ```

  V 7.5.0.4 **Propriedade do sistema com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS**

  Para clientes IBM WebSphere MQ classes for JMS e IBM WebSphere MQ classes for Java , configure a Java propriedade de sistema com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS para o valor TRUE para o aplicativo Java .

Por exemplo, é possível configurar a propriedade do sistema Java como uma opção `-D` quando o comando Java é chamado:

```
java -Dcom.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS=TRUE -cp <MQ_INSTALLATION_PATH>/java/lib/
com.ibm.mqjms.jar  my.java.applicationClass
```

Alternativamente, especifique a propriedade do sistema Java dentro de um arquivo de configuração do JMS, `jms.config`, se o aplicativo utilizar este arquivo.

V 7.5.0.5 **Propriedade `DisableClientAMS` no arquivo `mqclient.ini`**

Para clientes IBM WebSphere MQ classes for JMS e IBM WebSphere MQ classes for Java e para clientes C, inclua o nome da propriedade `DisableClientAMS` na sub-rotina **`Security`** do arquivo `mqclient.ini` , conforme mostrado no exemplo a seguir:

```
Security:
DisableClientAMS=Yes
```

Também é possível ativar o AMS, conforme mostrado no exemplo a seguir:

```
Security:
DisableClientAMS=No
```

Como proceder a seguir

Para obter mais informações sobre problemas com a abertura de filas protegidas do AMS, veja “Problemas ao abrir filas protegidas ao usar o JMS” na página 329.

Conceitos relacionados

“Intercepção de Agente do Canal de Mensagens (MCA)” na página 301
A intercepção do MCA permite que um gerenciador de filas em execução sob IBM WebSphere MQ ative seletivamente as políticas a serem aplicadas para os canais de conexão do servidor.

Tarefas relacionadas

Configurando um Cliente Usando um Arquivo de Configuração

Referências relacionadas

O arquivo de configuração IBM WebSphere MQ classes for JMS

Requisitos de certificado do AMS

Os certificados devem ter uma chave pública do RSA para que sejam usados com o Advanced Message Security.

Para obter mais informações sobre os diferentes tipos de chave pública e como criá-los, consulte “Certificados digitais e compatibilidade com CipherSpec em IBM WebSphere MQ” na página 35.

Extensões do uso da chave

As extensões de uso da chave colocam restrições adicionais na forma que um certificado pode ser usado.

No Advanced Message Security, o uso da chave deve ser configurado como a seguir: para certificados no padrão X.509 V3 ou mais recente que são usados para a qualidade de integridade da proteção, se as extensões de uso da chave estiverem configuradas, elas deverão incluir pelo menos um dos dois:

- **`nonRepudiation`**
- **`digitalSignature`**

Para a qualidade de proteção de privacidade, se as extensões de uso de chave estiverem configuradas, eles deverão também incluir a extensão do **`keyEncipherment`**.

Conceitos relacionados

“Qualidade de Proteção” na página 316
A proteção de políticas do Advanced Message Security implicam uma qualidade de proteção (QOP).

métodos de validação de certificado no IBM WebSphere MQ Advanced Message Security

É possível usar o IBM WebSphere MQ Advanced Message Security para detectar e rejeitar os certificados revogados para que as mensagens em suas filas não sejam protegidas usando certificados que não cumprem as normas de segurança.

O IBM WebSphere MQ AMS permite verificar a validade do certificado usando o Online Certificate Status Protocol (OCSP) ou a lista de revogação de certificados (CRL).

O IBM WebSphere MQ AMS pode ser configurado para a verificação de OCSP ou CRL ou ambos. Se ambos os métodos forem ativados, então, por motivos de desempenho, o IBM WebSphere MQ AMS usará o OCSP para o status de revogação primeiro. Se o status da revogação de um certificado for indeterminado após a verificação de OCSP, o IBM WebSphere MQ AMS usará a verificação de CRL.

Conceitos relacionados

"Online Certificate Status Protocol (OCSP)" na página 306
O Online Certificate Status Protocol (OCSP) determina se um certificado foi revogado e, portanto, ajuda a determinar se o certificado pode ser confiável.

"Listas de revogação de certificados (CRLs)" na página 308
As CRLs contêm uma lista de certificados que foram marcados pela Autoridade de certificação (CA) como não mais confiável por vários motivos, por exemplo, a chave privada foi perdida ou comprometida.

Online Certificate Status Protocol (OCSP)

O Online Certificate Status Protocol (OCSP) determina se um certificado foi revogado e, portanto, ajuda a determinar se o certificado pode ser confiável.

Ativando Verificação de OCSP em Interceptores Nativos

Para ativar a verificação de Online Certificate Status Protocol (OCSP) no Advanced Message Security, você deve modificar o arquivo de configuração de chaves.

Procedimento

Inclua as seguintes opções no arquivo de configuração de chaves :

Nota: Os valores para as opções individuais fornecidos na tabela são padrão.

Especifique um dos seguintes valores:

- `ocsp.enable=on`
- `ocsp.url=<`*`reposnder_URL`*`>`
- `ocsp.http.proxy.host=<`*`OCSP_proxy`*`>`

Opção	Descrição
`ocsp.enable=off`	Ative a verificação de OCSP se o certificado que está sendo verificado tiver um Authority Info Access Extension com um método de acesso PKIX_AD_OCSP que contém uma URI de onde o Respondente de OCSP está localizado. Valores possíveis: `on`/`off`.
`ocsp.url=<`*`reposnder_URL`*`>`	O endereço da URL do respondente do OCSP.
`ocsp.http.proxy.host=<`*`OCSP_proxy`*`>`	O endereço da URL do servidor proxy do OCSP.
`ocsp.http.proxy.port=<`*`port_number`*`>`	Número da porta do servidor proxy do OCSP
`ocsp.nonce.generation=on/off`	Gere nonce ao consultar o OCSP. O valor padrão é `off`.

Opção	Descrição
`ocsp.nonce.check=on/off`	Verifique o nonce após receber a resposta do OCSP. O valor padrão é `off`.
`ocsp.nonce.size=8`	Tamanho do nonce em bytes.
`ocsp.http.get=on/off`	Especifique HTTP GET como seu método de solicitação. Se essa opção estiver configurada como `off`, HTTP POST é utilizado.
`ocsp.max_response_size=20480`	Tamanho máximo de resposta do respondente do OCSP fornecido em bytes.
`ocsp.cache_size=100`	Ative cache de resposta do OCSP interno e configure o limite para o número de entradas de cache.
`ocsp.timeout=30`	Tempo de espera por uma resposta do servidor, em segundos, após o qual o Advanced Message Security atinge o tempo limite.

Ativando a verificação do OCSP em Java..
Para ativar o registro de entrada do OCSP para Java no Advanced Message Security, modifique o arquivo `java.security` ou o arquivo de configuração do keystore.

Sobre esta tarefa

Existem duas maneiras de ativar a verificação de OCSP no Advanced Message Security:

Usando java.security
Verifique se o seu certificado tem o Authority Information Access (AIA) configurado.

Procedimento

1. Se AIA não estiver configurado ou se você desejar substituir seu certificado, edite o arquivo `$JAVA_HOME/lib/security/java.security` com as seguintes propriedades:

   ```
   ocsp.responderURL=http://url.to.responder:port
   ocsp.responderCertSubjectName=CN=Example CA,O=IBM,C=US
   ```

   e ative a verificação de OCSP editando o arquivo `$JAVA_HOME/lib/security/java.security` com a linha a seguir:

   ```
   ocsp.enable=true
   ```

2. Se AIA estiver configurado, ative a verificação de OCSP editando o arquivo `$JAVA_HOME/lib/security/java.security` com a linha a seguir:

   ```
   ocsp.enable=true
   ```

Como proceder a seguir

Se você estiver usando o Java Security Manager, conclua a configuração, inclua a permissão Java a seguir em `lib/security/java.policy`

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
```

Procedimento

Inclua o atributo a seguir no arquivo de configuração:

```
ocsp.enable=true
```

Importante: A configuração desse atributo no arquivo de configuração substitui as configurações java.security.

Como proceder a seguir

Para concluir a configuração, inclua as seguintes permissões Java em `lib/security/java.policy`:

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
permission java.security.SecurityPermission "setProperty.ocsp.enable";
```

Listas de revogação de certificados (CRLs)

As CRLs contêm uma lista de certificados que foram marcados pela Autoridade de certificação (CA) como não mais confiável por vários motivos, por exemplo, a chave privada foi perdida ou comprometida.

Para validar certificados, o Advanced Message Security constrói uma cadeia de certificados que consiste no certificado do signatário e a cadeia de certificados da autoridade de certificação (CA) até uma âncora de confiança. Uma âncora de confiança é um arquivo keystore confiável que contém um certificado confiável ou um certificado de raiz confiável que é usado para assegurar a confiança de um certificado.IBM WebSphere MQ AMS Verifica o caminho do certificado usando um algoritmo de validação PKIX Quando a cadeia é criada e verificada, o IBM WebSphere MQ AMS conclui a validação de certificado que inclui a validação do problema e a data de expiração de cada certificado na cadeia em relação à data atual, verificando se a extensão do uso da chave está presente no certificado de Entidade Final. Se a extensão for anexada ao certificado, o IBM WebSphere MQ AMS verificará se o **digitalSignature** ou **nonRepudiation** também serão definidos. Se não forem, o `MQRC_SECURITY_ERROR` será relatado e registrado. Em seguida, o IBM WebSphere MQ AMS faz download das CRLs dos arquivos ou do LDAP, dependendo dos valores que foram especificados no arquivo de configuração. Somente as CRLs que são codificadas no formato DER são suportadas pelo IBM WebSphere MQ AMS. Se nenhuma configuração relacionada a CRL for localizada no arquivo de configuração do keystore, o IBM WebSphere MQ AMS não executará nenhuma verificação de validade da CRL. Para cada certificado de autoridade de certificação o IBM WebSphere MQ AMS consulta o LDAP para CRLs usando Nomes distintos de uma CA para localizar sua CRL. Os atributos a seguir são incluídos na consulta LDAP:

```
certificateRevocationList,
certificateRevocationList;binary,
authorityRevocationList,
authorityRevocationList;binary
deltaRevocationList
deltaRevocationList;binary,
```

Nota: `deltaRevocationList` é suportado somente quando ele for especificado como pontos de distribuição.

Ativando o suporte à validação do certificado e lista de revogação de certificado em interceptores nativos

É necessário modificar o arquivo de configuração do keystore para que o Advanced Message Security possa fazer download de CLRs do servidor Lightweight Directory Access Protocol (LDAP).

Procedimento

Inclua as opções a seguir no arquivo de configuração:

Nota: Os valores para as opções individuais fornecidos na tabela são padrão.

Opção	Descrição
`crl.ldap.host=<host_name>`	Nome do host do servidor LDAP.
`crl.ldap.port=<port_number>`	Número da porta do servidor LDAP. É possível especificar até 11 servidores. Vários hosts LDAP são usados para assegurar o failover transparente no caso de falha de conexão LDAP. É esperado que todos os servidores LDAP sejam réplicas e contenham os mesmos dados. Quando o interceptor Java do IBM WebSphere MQ AMS se conecta com êxito a um servidor LDAP, ele não tenta fazer download de CRLs dos servidores restantes fornecidos.
`crl.cdp=off`	Use essa opção para verificar ou usar extensões CRLDistributionPoints em certificados.
`crl.ldap.version=3`	Número da versão do protocolo LDAP. Valores possíveis: 2 ou 3.
`crl.ldap.user=cn=<username>`	Efetue login no servidor LDAP.. Se esse valor não for especificado, os atributos CRL no LDAP deverão ser legíveis no mundo
`crl.ldap.pass=<password>`	Senha para o servidor LDAP.
`crl.ldap.cache_lifetime=0`	Tempo de vida do cache de LDAP em segundos. Valores possíveis: 0-86400.
`crl.ldap.cache_size=50`	Tamanho do cache do LDAP. Esta opção pode ser especificada apenas se o valor `crl.ldap.cache_lifetime` for maior do que `0`.
`crl.http.proxy.host=some.host.com`	Porta do servidor proxy HTTP para recuperação do CDP CRL.
`crl.http.proxy.port=8080`	Número da porta do servidor proxy Http.
`crl.http.max_response_size=204800`	O tamanho máximo de CRL do , em bytes, que pode ser recuperado de um servidor HTTP que é aceito pelo GSKit.
`crl.http.timeout=30`	Tempo de espera para uma resposta do servidor, em segundos, após o tempo limite do IBM WebSphere MQ AMS .
`crl.http.cache_size=0`	Tamanho do cache HTTP, em bytes.

Ativando o suporte da lista de revogação de certificado em Java
Para ativar o suporte CRL no Advanced Message Security, deve-se modificar o arquivo de configuração de keystore para permitir que o IBM WebSphere MQ AMS faça download das CRLs para o servidor Lightweight Directory Access Protocol (LDAP) e configure o arquivo java.security.

Procedimento

1. Inclua as opções a seguir no arquivo de configuração:

Cabeçalho	Descrição
`crl.ldap.host=<host_name>`	Nome do host LDAP.

Cabeçalho	Descrição
`crl.ldap.port=<port_number>`	Número da porta do servidor LDAP. É possível especificar até 11 servidores. Vários hosts LDAP são usados para assegurar o failover transparente no caso de falha de conexão LDAP. É esperado que todos os servidores LDAP sejam réplicas e contenham os mesmos dados. Quando o interceptor Java do IBM WebSphere MQ AMS se conecta com êxito a um servidor LDAP, ele não tenta fazer download de CRLs dos servidores restantes fornecidos. Java não usa valores `crl.ldap.user` e crl.ldaworldp.pass . Ele não usa um usuário e senha ao se conectar a um servidor LDAP. Como consequência, os atributos de CRL LDAP devem ser legíveis mundialmente.
`crl.cdp=on/off`	Use essa opção para verificar ou usar extensões CRLDistributionPoints em certificados.

2. Modify the `JRE/lib/security/java.security` file with the following properties:

Nome da Propriedade	Descrição
`com.ibm.security.enableCRLDP`	Esta propriedade usa os valores a seguir: `true`, `false`. Se for configurada como `true`, ao fazer a verificação de revogação de certificado, as CRLs serão localizadas usando a URL da extensão de pontos de distribuição de CRL do certificado. Se for configurada como `false` ou não configurada, a verificação de CRL usando a extensão de pontos de distribuição de CRL será desativada.
`ibm.security.certpath.ldap.cache.lifetime`	Essa propriedade pode ser usada para configurar o tempo de vida de entradas no cache de memória de LDAP CertStore para um valor em segundos. Um valor de 0 desativa o cache; -1 significa tempo de vida ilimitado. Se não for configurado, o tempo de vida padrão será 30 segundos.

Nome da Propriedade	Descrição
com.ibm.security.enableAIAEXT	Esta propriedade usa os valores a seguir: true, false. Se for configurada como true, quaisquer extensões de Authority Information Access que forem encontradas dentro dos certificados do caminho do certificado que está sendo construído serão examinadas para determinar se elas contêm URIs LDAP. Para cada URI LDAP localizado, um objeto LDAPCertStore será criado e incluído na coleção de CertStores que é usada para localizar outros certificados que são requeridos para construir o caminho do certificado. Se for configurada como false ou não configurada, objetos LDAPCertStore adicionais não serão criados.

Protegendo as senhas no Java

Armazenamento de keystore e senhas de chave privada como texto simples representam um risco de segurança, então o Advanced Message Security fornece uma ferramenta que pode misturar essas senhas usando a chave do usuário, que está disponível no arquivo keystore.

Antes de começar

O proprietário do arquivo keystore.conf deve assegurar que somente o proprietário do arquivo esteja intitulado para ler o arquivo. A proteção de senhas descrita neste capítulo é somente uma medida de proteção adicional.

Procedimento

1. Edite o arquivo keystore.conf para incluir o caminho para o rótulo de keystore e usuários.

```
jceks.keystore = c:/Documents and Setting/Alice/AliceKeystore
jceks.certificate = AliceCert
jceks.provider = IBMJCE
```

2. Para executar a ferramenta, emita:

```
java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
private_key_password
```

   Uma saída com senhas criptografadas será gerada e poderá ser copiada para o arquivo keystore.conf.

   Para copiar a saída para o arquivo keystore.conf automaticamente, execute:

```
java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
private_key_password >> ~/<path_to_keystore>/keystore.conf
```

   Nota:

   Para uma lista de locais padrão do keystore.conf em várias plataformas, consulte “Usando keystores e certificados” na página 299.

Exemplo

Aqui está um exemplo de tal saída:

```
#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
```

Administrando políticas de segurança do IBM WebSphere MQ Advanced Message Security

O IBM WebSphere MQ Advanced Message Security usa políticas de segurança para especificar a criptografia criptográfica e os algoritmos de assinatura para criptografar e autenticar mensagens que fluem através das filas.

Visão geral das políticas de segurança

Políticas de segurança IBM Advanced Message Security são objetos conceituais que descrevem a maneira como uma mensagem é criptografada e assinada criptograficamente.

Para obter detalhes sobre atributos da política de segurança, consulte os subtópicos a seguir:

Conceitos relacionados

"Qualidade de Proteção" na página 316
A proteção de políticas do Advanced Message Security implicam uma qualidade de proteção (QOP).

"Atributos de política de segurança" na página 315
É possível usar o Advanced Message Security para selecionar um determinado algoritmo ou método para proteger os dados.

Nome da política

O nome da política é um nome exclusivo que identifica uma política específica do Advanced Message Security e a fila para a qual ela se aplica.

O nome da política deve ser o mesmo que o nome da fila para a qual ela deve ser aplicada. Há um mapeamento um-para-um entre uma política do Advanced Message Security (IBM WebSphere MQ AMS) e uma fila.

Ao criar uma política com o mesmo nome de uma fila, você ativa a política para essa fila. As filas sem correspondência de nomes de política não são protegidas pelo IBM WebSphere MQ AMS.

O escopo da política é relevante ao gerenciador de filas local e suas filas. Os gerenciadores de filas remotas devem ter suas próprias políticas definidas localmente para as filas que eles gerenciam.

Algoritmo de Assinatura

O algoritmo de assinatura indica o algoritmo que deve ser usado ao assinar mensagens de dados.

Valores válidos são:

- MD5
- SHA-1
- família: SHA-2
  - SHA256
  - SHA384 (comprimento da chave mínimo aceitável - 768 bits)
  - SHA512 (comprimento da chave mínimo aceitável - 768 bits)

Uma política que não especifica um algoritmo de assinatura ou especifica um algoritmo de NONE, implica que as mensagens colocadas na fila associada à política não são assinadas.

Nota: A qualidade de proteção usada para as funções put e get de mensagem deve corresponder. Se houver uma qualidade de política de incompatibilidade de proteção entre a fila e a mensagem na fila, a

mensagem não será aceita e será enviada à fila de manipulação de erros. Essa regra se aplica para ambas as filas locais e remotas.

Algoritmo de criptografia

O algoritmo de criptografia indica o algoritmo que deve ser usado ao criptografar mensagens de dados colocadas na fila associada à política.

Valores válidos são:

- RC2
- DES
- 3DES
- AES128
- AES256

Uma política que não especifica um algoritmo de criptografia ou especifica um algoritmo de NONE implica que mensagens colocadas na fila associada à política não serão criptografadas.

Observe que uma política que especifica um algoritmo de criptografia diferente de NONE também deve especificar pelo menos um DN de destinatário e um algoritmo de assinatura porque as mensagens criptografadas do Advanced Message Security também são assinadas.

Importante: A qualidade de proteção usada para as funções put e get de mensagem deve corresponder. Se houver uma qualidade de política de incompatibilidade de proteção entre a fila e a mensagem na fila, a mensagem não será aceita e será enviada à fila de manipulação de erros. Essa regra se aplica para ambas as filas locais e remotas.

Tolerância

O atributo de tolerância indica se a IBM Advanced Message Security pode aceitar mensagens sem política de segurança especificada.

Ao recuperar uma mensagem de uma fila com uma política para criptografar mensagens, se a mensagem não for criptografada, ela será retornada ao aplicativo de chamada. Valores válidos são:

0
Não (**padrão**)

1
Sim.

Uma política que não especifica um valor de tolerância ou especifica 0, significa que as mensagens colocadas na fila associada à política devem corresponder às regras de política.

A tolerância é opcional e existe para facilitar o roll-out de configuração, no qual as políticas foram aplicadas a filas, mas essas filas já contêm mensagens que não possuem uma política de segurança especificada.

Nomes distintos do emissor

Os nomes distintos (DNs) do emissor identificam usuários que estão autorizados a colocar mensagens em uma fila.

IBM Advanced Message Security (IBM WebSphere MQ AMS) não verifica se uma mensagem foi colocada em uma fila protegida por dados por um usuário válido até que a mensagem seja recuperada. Neste momento, se a política estipular um ou mais emissores válidos e o usuário que colocou a mensagem na fila não estiver na lista de emissores válidos, o IBM WebSphere MQ AMS retornará um erro ao aplicativo de obtenção e colocará a mensagem em sua fila de erros.

Uma política pode ter zero ou mais DNs do emissor especificado. Se nenhum DN de remetente for especificado para a política, qualquer usuário poderá colocar mensagens de dados protegidos na fila, contanto que o certificado de usuário seja confiável.

Nomes distintos do emissor têm o formato a seguir:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Todos os DNs devem estar em maiúsculas. Todos os identificadores de nome do componente no DN devem ser especificados na ordem mostrada na tabela a seguir:

Nome do Componente	Value
CN	O nome comum para o objeto deste DN, como um nome completo ou a finalidade de um dispositivo.
OU	A unidade dentro da organização com o qual o objeto do DN é afiliado, como uma divisão corporativa ou um nome do produto.
O	A organização com a qual o objeto do DN está afiliado, como uma corporação.
L	A localidade (cidade ou municipalidade) onde o objeto do DN está localizado.
Estado	O nome do estado ou da província onde o objeto do DN está localizado.
C	O país onde o objeto do nome distinto (DN) está localizado.

- Se um ou mais DNs de remetentes forem especificados para a política, somente aqueles usuários poderão colocar mensagens na fila associada com a política.
- Os DNs de remetentes, quando especificados, devem corresponder exatamente ao DN contido no certificado digital associado com o usuário que coloca a mensagem.
- O IBM WebSphere MQ AMS suporta DNs somente com valores do conjunto de caracteres Latin-1. Para criar DNs com caracteres do conjunto, deve-se primeiro criar um certificado com um DN que é criado na codificação UTF-8 usando plataformas UNIX com a codificação UTF-8 ativada ou com o utilitário iKeyman . Em seguida, deve-se criar uma política de uma plataforma UNIX com a codificação UTF-8 ativada ou usar o plug-in do IBM WebSphere MQ AMS para WebSphere MQ.

Conceitos relacionados

“Nomes distintos do destinatário” na página 314
Os nomes distintos (DN) do destinatário identificam usuários que estão autorizados a recuperar mensagens em uma fila.

Nomes distintos do destinatário

Os nomes distintos (DN) do destinatário identificam usuários que estão autorizados a recuperar mensagens em uma fila.

Uma política pode ter zero ou mais DNs do destinatário especificado. Os nomes distintos do destinatário têm o formato a seguir:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Todos os DNs devem estar em maiúsculas. Todos os identificadores de nome do componente no DN devem ser especificados na ordem mostrada na tabela a seguir:

Nome do Componente	Value
CN	O nome comum para o objeto deste DN, como um nome completo ou a finalidade de um dispositivo.

Nome do Componente	Value
OU	A unidade dentro da organização com o qual o objeto do DN é afiliado, como uma divisão corporativa ou um nome do produto.
O	A organização com a qual o objeto do DN está afiliado, como uma corporação.
L	A localidade (cidade ou municipalidade) onde o objeto do DN está localizado.
Estado	O nome do estado ou da província onde o objeto do DN está localizado.
C	O país onde o objeto do nome distinto (DN) está localizado.

- Se nenhum DN de destinatário for especificado para a política, qualquer usuário poderá receber mensagens da fila associada com a política.
- Se um ou mais DNs do destinatário forem especificados para a política, apenas esses usuários poderão obter mensagens da fila associada à política.
- Os DNs de destinatários, quando especificados, devem corresponder exatamente ao DN contido no certificado digital associado com o usuário que recebe a mensagem.
- O Advanced Message Security suporta DNs somente com valores do conjunto de caracteres Latin-1. Para criar DNs com caracteres do conjunto, deve-se primeiro criar um certificado com um DN que é criado na codificação UTF-8 usando plataformas UNIX com a codificação UTF-8 ativada ou com o utilitário iKeyman . Em seguida, deve-se criar uma política de uma plataforma UNIX com a codificação UTF-8 ativada ou usar o plug-in do Advanced Message Security para WebSphere MQ.

Conceitos relacionados

“Nomes distintos do emissor” na página 313
Os nomes distintos (DNs) do emissor identificam usuários que estão autorizados a colocar mensagens em uma fila.

Atributos de política de segurança

É possível usar o Advanced Message Security para selecionar um determinado algoritmo ou método para proteger os dados.

Uma política de segurança é um objeto conceitual que descreve a maneira como uma mensagem é criptograficamente criptografada e assinada. A tabela a seguir apresenta os atributos da política de segurança no Advanced Message Security:

Atributos	Descrição
Nome da política	Nome exclusivo da política para um gerenciador de filas.
Algoritmo de Assinatura	O algoritmo criptográfico que é usado para assinar mensagens antes do envio.
Algoritmo de criptografia	O algoritmo criptográfico que é usado para criptografar mensagens antes do envio.
Lista de destinatários	Lista de nomes distintos (DNs) de certificado dos destinatários em potencial de uma mensagem.
Lista de verificação do DN de assinatura	Lista de DNs de assinatura para serem validados durante a recuperação da mensagem.

No Advanced Message Security, as mensagens são criptografadas com uma chave simétrica e a chave simétrica é criptografada com as chaves públicas dos destinatários. Chaves públicas são criptografadas

com o algoritmo RSA, com chaves de um tamanho efetivo de até 2048 bits. A criptografia de chave assimétrica real depende do comprimento da chave do certificado.

Os algoritmos de chave simétrica suportados são os seguintes:

- RC2
- DES
- 3DES
- AES128
- AES256

O Advanced Message Security também suporta as funções hash de criptografia a seguir:

- MD5
- SHA-1
- família: SHA-2
  - SHA256
  - SHA384 (comprimento da chave mínimo aceitável - 768 bits)
  - SHA512 (comprimento da chave mínimo aceitável - 768 bits)

Nota: A qualidade de proteção usada para as funções put e get de mensagem deve corresponder. Se houver uma qualidade de política de incompatibilidade de proteção entre a fila e a mensagem na fila, a mensagem não será aceita e será enviada à fila de manipulação de erros. Essa regra se aplica para ambas as filas locais e remotas.

Qualidade de Proteção

A proteção de políticas do Advanced Message Security implicam uma qualidade de proteção (QOP).

A qualidade de três níveis de proteção no Advanced Message Security dependem dos algoritmos criptográficos que são usados para assinar e criptografar a mensagem:

- Privacidade - as mensagens colocadas na fila devem ser assinadas e criptografadas.
- Integridade - as mensagens colocadas na fila devem ser assinadas pelo emissor.
- Nenhum - Nenhuma proteção de dados é aplicável.

Uma política que determina que as mensagens devem ser assinadas quando colocadas em uma fila tem um QOP de INTEGRITY. Um QOP de INTEGRITY significa que uma política estipula um algoritmo de assinatura, mas não estipula um algoritmo de criptografia. As mensagens protegidas por integridade são também referidas como "ASSINADAS".

Uma política que determina que as mensagens devem ser assinadas e criptografadas quando colocadas em uma fila tem um QOP de PRIVACY. Um QOP de PRIVACIDADE significa que uma política estipula um algoritmo de assinatura e um algoritmo de criptografia. As mensagens protegidas por privacidade são também referidas como "SELADAS".

Uma política que não estipula um algoritmo de assinatura ou um algoritmo de criptografia tem um QOP de NONE. Advanced Message Security Não fornece proteção de dados para filas que tenham uma política com um QOP de NONE

Gerenciando Políticas de Segurança

Uma política de segurança é um objeto conceitual que descreve a maneira como uma mensagem é criptograficamente criptografada e assinada.

Todas as tarefas administrativas relacionadas a políticas de segurança são executadas por meio do local a seguir:

- Em plataformas UNIX : `<MQInstallRoot>/bin`
- Nas plataformas Windows , as tarefas administrativas podem ser executadas a partir de qualquer local conforme a variável de ambiente PATH é atualizada na instalação.

Tarefas relacionadas

“Criação de políticas de segurança” na página 317
As políticas de segurança definem a maneira na qual uma mensagem será protegida quando a mensagem for colocada ou como uma mensagem deve ter sido protegida quando uma mensagem é recebida.

“Mudando as políticas de segurança” na página 318
É possível usar o Advanced Message Security para alterar detalhes de políticas de segurança que você já definiu.

“Exibindo e realizando dump das políticas de segurança” na página 318
Use o comando `dspmqspl` para exibir uma lista de todas as políticas de segurança ou detalhes de uma política denominada dependendo dos parâmetros da linha de comandos que você fornecer.

“Removendo as políticas de segurança” na página 319
Para remover políticas de segurança no Advanced Message Security, deve-se usar o comando `setmqspl`.

Criação de políticas de segurança

As políticas de segurança definem a maneira na qual uma mensagem será protegida quando a mensagem for colocada ou como uma mensagem deve ter sido protegida quando uma mensagem é recebida.

Antes de começar

Há algumas condições de entrada que devem ser atendidas ao criar políticas de segurança:

- O gerenciador de filas deve estar em execução.
- O nome de uma política de segurança deve seguir Regras para nomear objetos do WebSphere MQ.
- Você deve ter as autoridades `+connect +inq +chg` necessárias para criar uma política de segurança. Para obter a sintaxe completa do comando de alteração de autorização, consulte **setmqaut**
- Certifique-se de ter as permissões necessárias para operar nas filas e gerenciadores de filas do WebSphere MQ . Para obter mais informações, consulte “Concedendo permissões OAM” na página 321

Exemplo

A seguir há um exemplo da criação de uma política no gerenciador de filas `QMGR`. A política especifica que as mensagens sejam assinadas usando o algoritmo SHA1 e criptografadas usando o algoritmo AES256 para certificados com DN: CN=joe,O=IBM,C=US e DN: CN=jane,O=IBM,C=US. Essa política está conectada ao `MY.QUEUE`:

```
$ setmqspl -m QMGR -p MY.QUEUE -s SHA1 -e AES256 -r CN=joe,O=IBM,C=US -r CN=jane,O=IBM,C=US
```

Aqui está um exemplo de criação de política no gerenciador de filas `QMGR`. A política especifica que as mensagens sejam criptografadas usando o algoritmo DES para certificados com DNs: CN=john,O=IBM,C=US e CN=jeff,O=IBM,C=US e assinadas com o algoritmo `MD5` para certificados com DN: CN=phil,O=IBM,C=US

```
$ setmqspl -m QMGR -p MY.OTHER.QUEUE -s MD5 -e DES -r CN=john,O=IBM,C=US -r CN=jeff,O=IBM,C=US
-a CN=phil,O=IBM,C=US
```

Nota:

- A qualidade da proteção sendo usada para a mensagem de colocação e obtenção deve corresponder. Se a qualidade da política de proteção que é definida para a mensagem for mais fraca do que a definida para uma fila, a mensagem será enviada para a fila de manipulação de erros. Esta política é válida para as filas local e remota.

Referências relacionadas

Lista completa dos atributos do comando setmqspl

Mudando as políticas de segurança

É possível usar o Advanced Message Security para alterar detalhes de políticas de segurança que você já definiu.

Antes de começar

- O gerenciador de filas no qual deseja operar deve estar em execução.
- Deve-se ter autoridades `+connect +inq +chg` necessárias para criar políticas de segurança. Para obter a sintaxe completa do comando de alteração de autorização, consulte **setmqaut**

Sobre esta tarefa

Para mudar as políticas de segurança, aplique o comando `setmqspl` a uma política já existente fornecendo novos atributos.

Exemplo

Aqui está um exemplo de criação de uma política denominada MYQUEUE em um gerenciador de filas denominado QMGR especificando que as mensagens serão criptografadas usando o algoritmo RC2 para certificados com DN:CN=bob,O=IBM,C=US e assinados com o algoritmo SHA1 para certificados com DN:CN=jeff,O=IBM,C=US.

```
setmqspl -m QMGR -p MYQUEUE -e RC2 -s SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Para alterar essa política, emita o comando `setmqspl` com todos os atributos do exemplo, mudando somente os valores que você deseja modificar. Neste exemplo, a política criada anteriormente é conectada a uma nova fila e seu algoritmo de criptografia é mudado para AES256:

```
setmqspl -m QMGR -p MYQUEUE -e AES256 -s SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Referências relacionadas

setmqspl

Exibindo e realizando dump das políticas de segurança

Use o comando `dspmqspl` para exibir uma lista de todas as políticas de segurança ou detalhes de uma política denominada dependendo dos parâmetros da linha de comandos que você fornecer.

Antes de começar

- Para exibir detalhes de políticas de segurança, o gerenciador de filas deve existir e estar em execução.
- Você deve ter permissões `+connect +inq +dsp` necessárias aplicadas a um gerenciador de filas para exibir e fazer dump de políticas de segurança. Para obter a sintaxe completa do comando de alteração de autorização, consulte **setmqaut**

Sobre esta tarefa

Aqui está a lista de sinalizadores de comando `dspmqspl`:

Tabela 27. Sinalizadores de comando `dspmqspl`.

Sinalizador de comando	Explanation
`-m`	Nome do gerenciador de filas (**obrigatório**).
`-p`	Nome da política.
`-export`	Incluir este sinalizador gera uma saída que pode ser facilmente aplicada a um gerenciador de filas diferente.

Exemplo

Neste exemplo, criaremos duas políticas de segurança para `venus.queue.manager`:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s MD5 -a "CN=signer1,O=IBM,C=US" -e NONE
setmqspl -m venus.queue.manager -p AMS_POL_06_THREE -s MD5 -a "CN=another signer,O=IBM,C=US" -e
NONE
```

Este exemplo mostra um comando que exibe detalhes de todas as políticas definidas para `venus.queue.manager` e a saída que ele produz:

```
dspmqspl -m  venus.queue.manager

Policy Details:
Policy name: AMS_POL_04_ONE
Quality of protection: INTEGRITY
Signature algorithm: MD5
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: MD5
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

Este exemplo mostra um comando que exibe detalhes de uma política de segurança selecionada definida para `venus.queue.manager` e a saída que ela produz:

```
dspmqspl -m venus.queue.manager -p AMS_POL_06_THREE

Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: MD5
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

No próximo exemplo, em primeiro lugar, criamos uma política de segurança e, em seguida, exportamos a política usando o sinalizador `-export`:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s MD5 -a "CN=signer1,O=IBM,C=US" -e NONE

dspmqspl -m venus.queue.manager -export > policies.[bat|sh]
```

Para importar uma política de segurança:

- Em plataformas Windows , execute `policies.bat`
- Nas plataformas UNIX :
  1. Efetue logon como um usuário que pertence ao grupo de administração `mqm` WebSphere MQ.
  2. Emita `. policies.sh`..

Referências relacionadas

Lista completa dos atributos do comando dspmqspl

Removendo as políticas de segurança

Para remover políticas de segurança no Advanced Message Security, deve-se usar o comando `setmqspl`.

Antes de começar

Há algumas condições de entrada que devem ser atendidas ao gerenciar políticas de segurança:

- O gerenciador de filas deve estar em execução.
- Deve-se ter autoridades `+connect +inq +chg` necessárias para criar políticas de segurança. Para obter a sintaxe completa do comando de alteração de autorização, consulte **setmqaut**

Sobre esta tarefa

Use o comando `setmqspl` com a opção `-remove`.

Exemplo

Aqui está um exemplo de remoção de uma política:

```
$ setmqspl -m QMGR -remove -p MY.OTHER.QUEUE
```

Referências relacionadas

Lista completa dos atributos do comando setmqspl

Proteção da fila do sistema

As filas do sistema ativam a comunicação entre WebSphere MQ e seus aplicativos auxiliares. Sempre que um gerenciador de filas é criado, uma fila do sistema também é criada para armazenar mensagens internas e dados das mensagens do WebSphere MQ. É possível proteger filas do sistema com o Advanced Message Security para que somente usuários autorizados possam acessá-las ou decriptografá-las.

A proteção de fila do sistema segue o mesmo padrão que a proteção das filas regulares. Consulte “Criação de políticas de segurança” na página 317.

Para usar a proteção de fila do sistema em plataformas Windows , copie o arquivo `keystore.conf` para o seguinte diretório:

```
c:\Documents and Settings\Default User\.mqs\keystore.conf
```

Para fornecer proteção para o `SYSTEM.ADMIN.COMMAND.QUEUE`, o servidor de comandos deve ter acesso ao `keystore` e ao `keystore.conf`, que contêm chaves e uma configuração para que o servidor de comandos possa acessar chaves e certificados. Todas as mudanças feitas na política de segurança do `SYSTEM.ADMIN.COMMAND.QUEUE` requerem a reinicialização de um servidor de comando

Todas as mensagens que são enviadas e recebidas da fila de comandos são assinadas ou assinadas e criptografadas dependendo das configurações de política. Se um administrador definir assinantes autorizados, as mensagens de comandos que não passam pela verificação de Nome Distinto (DN) do assinante não serão executadas pelo servidor de comandos e não serão roteadas para a fila de manipulação de erros do Advanced Message Security As mensagens enviadas como respostas para as filas dinâmicas temporárias do WebSphere MQ Explorer não são protegidas pelo AMS do WebSphere MQ .

Mudanças nas políticas de segurança do Advanced Message Security requerem que você reinicie o servidor de comando do WebSphere MQ

As políticas de segurança não têm efeito sobre as filas SYSTEM a seguir:

- SYSTEM.ADMIN.ACCOUNTING.QUEUE
- SYSTEM.ADMIN.ACTIVITY.QUEUE
- SYSTEM.ADMIN.CHANNEL.EVENT
- SYSTEM.ADMIN.COMMAND.EVENT
- 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
- SYSTEM.BROKER.CONTROL.QUEUE
- SYSTEM.BROKER.DEFAULT.STREAM
- SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS
- SYSTEM.CHANNEL.INITQ
- SYSTEM.CHANNEL.SYNCQ
- SYSTEM.CICS.INITIATION.QUEUE
- SYSTEM.CLUSTER.COMMAND.QUEUE
- SYSTEM.CLUSTER.HISTORY.QUEUE
- SYSTEM.CLUSTER.REPOSITORY.QUEUE
- SYSTEM.CLUSTER.TRANSMIT.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
- SYSTEM.PENDING.DATA.QUEUE
- SYSTEM.PROTECTION.ERROR.QUEUE
- SYSTEM.PROTECTION.POLICY.QUEUE
- SYSTEM.RETAINED.PUB.QUEUE
- SYSTEM.SELECTION.EVALUATION.QUEUE
- SYSTEM.SELECTION.VALIDATION.QUEUE

Concedendo permissões OAM

As permissões de arquivo autorizam todos os usuários a executar os comandos `setmqspl` e `dspmqspl`. No entanto, o IBM Advanced Message Security depende do Object Authority Manager (OAM) e cada tentativa de executar esses comandos por um usuário que não pertence ao grupo mqm, que é o grupo de administração do WebSphere MQ ou não tem permissões para ler as configurações de política de segurança que são concedidas, resulta em um erro.

Procedimento

Para conceder as permissões necessárias para um usuário, execute:

```
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
```

Eventos de comando e configuração

Com o Advanced Message Security, é possível gerar mensagens de eventos de comando e configuração, que podem ser registradas e servir como um registro das mudanças da política para auditoria.

Eventos de comando e configuração gerados pelo WebSphere MQ são mensagens do formato PCF enviadas para filas dedicadas.

Mensagens de eventos de configuração são enviadas ao SYSTEM.ADMIN.CONFIG.EVENT no gerenciador de filas no qual o evento ocorre.

As mensagens de eventos de comando são enviadas para o SISTEMA SYSTEM.ADMIN.COMMAND.EVENT no gerenciador de filas no qual o evento ocorre.

Os eventos são gerados independentemente de ferramentas que você está usando para gerenciar as políticas de segurança do Advanced Message Security .

No Advanced Message Security, há quatro tipos de eventos gerados por diferentes ações em políticas de segurança:

- "Criação de políticas de segurança" na página 317, que geram duas mensagens do evento do WebSphere MQ:
  - Um evento de configuração
  - Um evento de comando
- "Mudando as políticas de segurança" na página 318, que gera três mensagens do evento do WebSphere MQ:
  - Um evento de configuração que contém os antigos valores de política de segurança
  - Um evento de configuração que contém os novos valores de política de segurança
  - Um evento de comando
- "Exibindo e realizando dump das políticas de segurança" na página 318, que gera uma mensagem do evento do WebSphere MQ:
  - Um evento de comando
- "Removendo as políticas de segurança" na página 319, que gera duas mensagens do evento do WebSphere MQ:
  - Um evento de configuração
  - Um evento de comando

Ativando e desativando a criação de log de eventos

Você controla os eventos de comando e configuração usando os atributos CONFIGEV e CMDEV do gerenciador de filas. Para ativar esses eventos, configure o atributo do gerenciador de filas apropriado para ENABLED. Para desativar esses eventos, configure o atributo apropriado do gerenciador de filas para DISABLED.

Procedimento

Eventos de Configuração

Para ativar eventos de configuração, configure CONFIGEV como ENABLED. Para desativar eventos de configuração, configure CONFIGEV para DISABLED. Por exemplo, é possível ativar eventos de configuração usando o comando MQSC a seguir:

```
ALTER QMGR CONFIGEV (ENABLED)
```

Eventos de Comando

Para ativar eventos de comando, configure CMDEV como ENABLED. Para ativar os eventos de comando para os comandos, exceto comandos DISPLAY MQSC e comandos Inquire PCF, configure o CMDEV para NODISPLAY. Para desativar eventos de comando, configure CMDEV para DISABLED. Por exemplo, é possível ativar eventos de comandos usando o comando MQSC a seguir:

```
ALTER QMGR CMDEV (ENABLED)
```

Tarefas relacionadas

Controlando a configuração, comando e eventos do criador de logs no Websphere MQ

Formato da mensagem do evento de comando

A mensagem do evento de comando consiste em estrutura MQCFH e os parâmetros PCF a seguir.

Aqui estão os valores MQCFH selecionados:

```
Type = MQCFT_EVENT;
Command = MQCMD_COMMAND_EVENT;
MsgSeqNumber = 1;
Control = MQCFC_LAST;
ParameterCount = 2;
CompCode = MQCC_WARNING;
Reason = MQRC_COMMAND_PCF;
```

Nota: O valor de ParameterCount é dois porque é sempre existem dois parâmetros do tipo MQCFGR (grupo). Cada grupo é constituído de parâmetros apropriados. Os dados do evento consistem em dois grupos, CommandContext e CommandData.

CommandContext contém:

EventUserID

Descrição :	O ID do usuário que emitiu o comando ou a chamada que gerou o evento. (Este é o mesmo ID do usuário que será usado para verificar a autoridade para emitir o comando ou chamada; para comandos recebidos de uma fila, este também será o identificador do usuário (UserIdentifier) do MD da mensagem de comando).
Identificador	MQCACF_EVENT_USER_ID.
Tipo de dado:	MQCFST.
Comprimento Máximo:	MQ_USER_ID_LENGTH.
Retornado:	Sempre.

EventOrigin

Descrição :	A origem da ação que causou o evento.
Identificador	MQIACF_EVENT_ORIGIN.
Tipo de dado:	MQCFIN.
Valores:	**MQEVO_CONSOLE** Comando de console - linha de comandos. **MQEVO_MSG** Mensagem de comando a partir do plug-in do WebSphere MQ Explorer.
Retornado:	Sempre.

EventQMgr

Descrição :	O gerenciador de filas no qual o comando ou chamada foi inserido. (O gerenciador de filas no qual o comando é executado e que gera o evento está no MD da mensagem do evento).
Identificador	MQCACF_EVENT_Q_MGR.
Tipo de dado:	MQCFST.
Comprimento Máximo:	MQ_Q_MGR_NAME_LENGTH.

Retornado: Sempre.

EventAccountingToken

Descrição : Para comandos recebidos como uma mensagem (MQEVO_MSG), o token de conta (AccountingToken) do MD da mensagem de comando.

Identificador MQBACF_EVENT_ACCOUNTING_TOKEN.

Tipo de dado: MQCFBS.

Comprimento Máximo: MQ_ACCOUNTING_TOKEN_LENGTH.

Retornado: Somente se EventOrigin for MQEVO_MSG.

EventIdentityData

Descrição : Para comandos recebidos como uma mensagem (MQEVO_MSG), os dados de identificação do aplicativo (ApplIdentityData) do MD da mensagem de comando.

Identificador MQCACF_EVENT_APPL_IDENTITY.

Tipo de dado: MQCFST.

Comprimento Máximo: MQ_APPL_IDENTITY_DATA_LENGTH.

Retornado: Somente se EventOrigin for MQEVO_MSG.

EventApplType

Descrição : Para comandos recebidos como uma mensagem (MQEVO_MSG), o tipo do aplicativo (PutApplType) do MD da mensagem de comando.

Identificador MQIACF_EVENT_APPL_TYPE.

Tipo de dado: MQCFIN.

Retornado: Somente se EventOrigin for MQEVO_MSG.

EventApplName

Descrição : Para comandos recebidos como uma mensagem (MQEVO_MSG), o nome do aplicativo (PutApplName) do MD da mensagem de comando.

Identificador MQCACF_EVENT_APPL_NAME.

Tipo de dado: MQCFST.

Comprimento Máximo: MQ_APPL_NAME_LENGTH.

Retornado: Somente se EventOrigin for MQEVO_MSG.

EventApplOrigin

Descrição : Para comandos recebidos como uma mensagem (MQEVO_MSG), os dados de origem do aplicativo (ApplOriginData) do MD da mensagem de comando.

Identificador MQCACF_EVENT_APPL_ORIGIN.

Tipo de dado: MQCFST.

Comprimento Máximo: MQ_APPL_ORIGIN_DATA_LENGTH.

Retornado: Somente se EventOrigin for MQEVO_MSG.

Command

Descrição :	O código de comando.
Identificador	MQIACF_COMMAND.
Tipo de dado:	MQCFIN.
Valores:	**MQCMD_INQUIRE_PROT_POLICY valor numérico 205** **MQCMD_CREATE_PROT_POLICY valor numérico 206** **MQCMD_DELETE_PROT_POLICY valor numérico 207** **MQCMD_CHANGE_PROT_POLICY valor numérico 208** Eles são definidos em WebSphere MQ 7.5 `cmqcfc.h` ..
Retornado:	Sempre.

CommandData contém elementos PCF que compõem o comando PCF.

Formato da mensagem do evento de configuração

Os eventos de configuração são mensagens PCF de formato padrão do Advanced Message Security.

Para obter os valores possíveis para o descritor de mensagens MQMD, consulte Mensagem de evento MQMD (descritor de mensagem).

Aqui estão os valores MQMD selecionados:

```
Format = MQFMT_EVENT
Peristence = MQPER_PERSISTENCE_AS_Q_DEF
PutApplType = MQAT_QMGR                      //for both CLI and command server
```

O buffer de mensagem consiste na estrutura MQCFH e na estrutura de parâmetros que o segue Para valores MQCFH possíveis, consulte Mensagem do evento MQCFH (cabeçalho PCF).

Aqui estão os valores MQCFH selecionados:

```
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}
```

Os parâmetros seguindo MQCFH são:

EventUserID

Descrição :	O ID do usuário que emitiu o comando ou a chamada que gerou o evento. (Este é o mesmo ID do usuário que será usado para verificar a autoridade para emitir o comando ou chamada; para comandos recebidos de uma fila, este também será o identificador do usuário (UserIdentifier) do MD da mensagem de comando).
Identificador	**MQCACF_EVENT_USER_ID**
Tipo de dado:	MQCFST.
Comprimento Máximo:	MQ_USER_ID_LENGTH.
Retornado:	Sempre.

SecurityId

Descrição : Valor de MQMD.AccountingToken no caso de mensagem do servidor de comando ou SID do Windows para comando local.

Identificador **MQBACF_EVENT_SECURITY_ID**

Tipo de dado: MQCBS.

Comprimento Máximo: MQ_SECURITY_ID_LENGTH.

Retornado: Sempre.

EventOrigin

Descrição : A origem da ação que causou o evento.

Identificador **MQIACF_EVENT_ORIGIN**

Tipo de dado: MQCFIN.

Valores:

MQEVO_CONSOLE
Comando de console - linha de comandos.

MQEVO_MSG
Mensagem de comando a partir do plug-in do WebSphere MQ Explorer.

Retornado: Sempre.

EventQMgr

Descrição : O gerenciador de filas no qual o comando ou chamada foi inserido. (O gerenciador de filas no qual o comando é executado e que gera o evento está no MD da mensagem do evento).

Identificador **MQCACF_EVENT_Q_MGR**

Tipo de dado: MQCFST

Comprimento Máximo: MQ_Q_MGR_NAME_LENGTH

Retornado: Sempre.

ObjectType

Descrição : Tipo de objeto.

Identificador **MQIACF_OBJECT_TYPE**

Tipo de dado: MQCFIN

Valor:

MQOT_PROT_POLICY
Política de proteção do Advanced Message Security **1019** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h` .

Retornado: Sempre.

PolicyName

Descrição : O nome da política do Advanced Message Security

Identificador **MQCA_POLICY_NAME**.

Tipo de dado: MQCFST.

Valor: **2112** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Comprimento Máximo: MQ_OBJECT_NAME_LENGTH.

Retornado: Sempre.

PolicyVersion

Descrição : A versão da política do Advanced Message Security

Identificador **MQIA_POLICY_VERSION**

Tipo de dado: MQCFIN

Value **238** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Retornado: Sempre

TolerateFlag

Descrição : O sinalizador de tolerância de política do Advanced Message Security.

Identificador **MQIA_TOLERATE_UNPROTECTED**

Tipo de dado: MQCFIN

Value **235** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Retornado: Sempre.

SignatureAlgorithm

Descrição : O algoritmo de assinatura da política do Advanced Message Security.

Identificador **MQIA_SIGNATURE_ALGORITHM**

Tipo de dado: MQCFIN

Valor: **236** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Retornado: Sempre que houver um algoritmo de assinatura definido na política do Advanced Message Security

EncryptionAlgorithm

Descrição : O algoritmo de criptografia da política do Advanced Message Security.

Identificador **MQIA_ENCRYPTION_ALGORITHM**

Tipo de dado: MQCFIN

Valor: **237** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Retornado: Sempre que houver um algoritmo de criptografia definido na política do WebSphere MQ

SignerDNs

Descrição : Assunto DistinguishedName dos assinantes permitidos.

Identificador **MQCA_SIGNER_DN**

Tipo de dado: MQCFSL

Valor: **2113** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h`.

Comprimento Máximo:	Maior DN de assinante na política, mas não maior que MQ_DISTINGUISHED_NAME_LENGTH
Retornado:	Sempre definido na política WebSphere MQ.

RecipientDNs

Descrição :	Assunto DistinguishedName dos assinantes permitidos.
Identificador	**MQCA_RECIPIENT_DN**
Tipo de dado:	MQCFSL
Valor:	**2114** -um valor numérico definido em WebSphere MQ 7.5 ou no arquivo `cmqc.h` .
Comprimento Máximo:	Maior DN do destinatário na política, mas não maior que MQ_DISTINGUISHED_NAME_LENGTH.
Retornado:	Sempre definido na política WebSphere MQ.

Problemas e Soluções

Esta seção descreve como resolver problemas que possam surgir com qualquer instalação do IBM Use estas informações para identificar e resolver quaisquer problemas relacionados ao Advanced Message Security

com.ibm.security.pkcsutil.PKCSException: Erro ao criptografar o conteúdo

O erro `com.ibm.security.pkcsutil.PKCSException: Error encrypting contents` sugere que IBM Advanced Message Security tem problemas ao acessar algoritmos criptográficos.

Se o erro a seguir for retornado pelo Advanced Message Security:

```
DRQJP0103E   The IBM WebSphere MQ Advanced Message Security Java interceptor failed to protect
message.
  com.ibm.security.pkcsutil.PKCSException: Error encrypting contents
(java.security.InvalidKeyException: Illegal key size or default parameters)
```

verifique se a política de segurança JCE em `JAVA_HOME/lib/security/local_policy.jar/*.policy` concede acesso aos algoritmos de assinatura usados na política do MQ AMS.

Se o algoritmo de assinatura que você deseja usar não for especificado em sua política de segurança atual, faça download do arquivo de políticas Java correto a partir dos seguintes locais:

- IBM Arquivos de Políticas SDK para Java 1.4.2.
- IBM Arquivos de Políticas do SDK para Java 5.0.
- IBM Arquivos de Políticas de SDK para Java 6.0.
- IBM Arquivos de Políticas do SDK para Java 7.0.

suporte de OSGi

Para usar o pacote configurável OSGi com IBM Advanced Message Security parâmetros adicionais são necessários.

Execute o parâmetro a seguir durante a inicialização do pacote configurável OSGi:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7
```

Ao usar a senha criptografada em seu keystore.conf, a instrução a seguir deve ser incluída quando pacote configurável OSGi estiver em execução:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7,com.ibm.misc
```

Restrição: O IBM WebSphere MQ AMS suporta comunicação usando apenas Classes Java Base do MQ para filas protegidas de dentro do pacote configurável OSGi.

Problemas ao abrir filas protegidas ao usar o JMS

Vários problemas podem surgir quando você abrir filas protegidas ao usar o IBM WebSphere MQ Advanced Message Security.

Você está executando o JMS e você recebe o erro 2085 (MQRC_UNKNOWN_OBJECT_NAME) juntamente com erro JMSMQ2008.

Você verificou que configurou o IBM WebSphere MQ Advanced Message Security conforme descrito em "Guia de iniciação rápida para clientes Java" na página 289.

Uma causa possível é que você está usando um não IBM Java Runtime Environment. Esta é uma limitação conhecida descrita em "Limitações conhecidas" na página 277.

Você não configurou a variável de ambiente AMQ_DISABLE_CLIENT_AMS.

Resolvendo o problema

Existem quatro opções para resolver este problema:

1. Inicie seu aplicativo JMS sob um IBM Java Runtime Environment (JRE) suportado.
2. Mova seu aplicativo para a mesma máquina em que o gerenciador de filas está em execução e conecte-o usando uma conexão de modo de ligações.

   Uma conexão de modo de ligações usa as bibliotecas nativas da plataforma para executar as chamadas API do IBM WebSphere MQ. Assim, o interceptor AMS nativo é usado para executar as operações AMS e não há confiança na capacidade do JRE.
3. Use um interceptor MCA, porque isto permite a assinatura e criptografia de mensagens assim que elas chegam no gerenciador de filas, sem a necessidade de o cliente executar qualquer processamento AMS.

   Dado que a proteção é aplicada no gerenciador de filas, um mecanismo alternativo deve ser usado para proteger as mensagens em trânsito do cliente para o gerenciador de filas. Mais comumente isso é alcançado configurando a criptografia SSL/TLS no canal de conexão do servidor usado pelo aplicativo.
4. Configure a variável de ambiente AMQ_DISABLE_CLIENT_AMS se você não desejar usar o IBM WebSphere MQ Advanced Message Security.

Consulte "Intercepção de Agente do Canal de Mensagens (MCA)" na página 301 para obter informações adicionais.

Nota: Uma política de segurança deve estar em prática para cada fila para as quais o MCA Interceptor entregar mensagens. Em outras palavras, a fila de destino precisa ter uma política de segurança AMS em prática com o nome distinto (DN) do assinante e destinatário correspondendo ao certificado atribuído ao MCA Interceptor. Ou seja, o DN do certificado designado pela propriedade `cms.certificate.channel.SYSTEM.DEF.SVRCONN` no `keystore.conf` usado pelo gerenciador de filas.

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte seu representante local do IBM para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a um IBM produto, programa ou serviço não se destina a estado ou significa que apenas esse produto IBM, programas ou serviços possam ser utilizados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM poderá ser utilizado em substituição. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não IBM são de responsabilidade do Cliente.

A IBM pode ter patentes ou aplicativos de patentes pendentes relativas aos assuntos tratados nesta publicação. O fornecimento desta publicação não garante ao Cliente nenhum sobre tais patentes. é possível enviar pedidos de licença, por escrito, para:

Relações Comerciais e Industriais da IBM
Av. Pasteur, 138-146
Botafogo
Rio, RJ 10504-1785
U.S.A.

Para pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

licença de propriedade intelectual
IBM World Trade Asia Corporation Licensing
IBM Japan, Ltd.
Minato-ku
Tóquio 103-8510, Japão

O parágrafo a seguir não se aplica a nenhum país em que tais disposições não estejam de acordo com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. Periodicamente, são feitas nas informações aqui contidas; essas alterações serão incorporadas em futuras edições desta publicação. IBM pode aperfeiçoar e/ou alterar no produto(s) e/ou programa(s) descritos nesta publicação a qualquer momento sem aviso prévio.

Referências nestas informações a websites não IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses websites. Os materiais contidos nesses websites não fazem parte dos materiais desse produto IBM e a utilização desses websites é de inteira responsabilidade do Cliente.

A IBM pode utilizar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

Av. Pasteur, 138-146
Av. Pasteur, 138-146

Botafogo
Rio de Janeiro, RJ
U.S.A.

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriadas, incluindo em alguns casos o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo o material licenciado disponível para ele são fornecidos pela IBM sob os termos do IBM Customer Agreement, IBM Contrato de Licença do Programa Internacional ou qualquer contrato equivalente entre as partes.

Todos os dados de desempenho aqui contidos foram determinados em um ambiente controlado. Portanto, os resultados obtidos em outros ambientes operacionais podem variar significativamente. Algumas medidas podem ter sido tomadas em sistemas em nível de desenvolvimento e não há garantia de que estas medidas serão iguais em sistemas geralmente disponíveis. Além disto, algumas medidas podem ter sido estimadas através de extrapolação. Os resultados reais podem variar. usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

As informações relativas a produtos não IBM foram obtidas junto aos fornecedores dos respectivos produtos, de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade nem qualquer outra reivindicação relacionada a produtos não IBM. Dúvidas sobre os recursos de produtos não IBM devem ser encaminhadas diretamente a seus fornecedores.

Todas as declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a alterações ou cancelamento sem aviso prévio e representam somente metas e objetivos.

Essas informações contêm exemplos de dados e relatórios utilizados em operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos incluem nomes de indivíduos, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com os nomes e endereços utilizados por uma empresa real é mera coincidência.

LICENÇA DE COPYRIGHT :

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, uso, marketing ou distribuição de programas aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de amostra são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a confiabilidade, manutenção ou função destes programas.

Se estiver visualizando estas informações em formato eletrônico, as fotografias e ilustrações coloridas poderão não aparecer.

Informações sobre a Interface de Programação

As informações da interface de programação, se fornecidas, destinam-se a ajudá-lo a criar software aplicativo para uso com este programa.

Este manual contém informações sobre interfaces de programação desejadas que permitem que o cliente grave programas para obter os serviços do IBM WebSphere MQ.

No entanto, estas informações também podem conter informações sobre diagnósticos, modificações e ajustes. As informações sobre diagnósticos, modificações e ajustes são fornecidas para ajudá-lo a depurar seu software aplicativo.

Importante: Não use essas informações de diagnóstico, modificação e ajuste como uma interface de programação, pois elas estão sujeitas a mudanças

Marcas comerciais

IBM, o logotipo IBM , ibm.com, são marcas registradas da IBM Corporation, registradas em várias jurisdições no mundo todo Uma lista atual de marcas registradas da IBM está disponível na Web em "Informações de copyright e marca registrada"www.ibm.com/legal/copytrade.shtml. Outros nomes de produtos e serviços podem ser marcas comerciais da IBM ou de outras empresas.

Microsoft e Windows são marcas comerciais da Microsoft Corporation nos Estados Unidos e/ou em outros países.

UNIX é uma marca registrada do The Open Group nos Estados Unidos e em outros países.

Linux é uma marca registrada de Linus Torvalds nos Estados Unidos e/ou em outros países.

Este produto inclui software desenvolvido pelo Projeto Eclipse (http://www.eclipse.org/).

Java e todas as marcas comerciais e logotipos baseados em Java são marcas comerciais ou marcas registradas da Oracle e/ou de suas afiliadas.

Part Number:

(1P) P/N: