9.1

Protección de IBM MQ

Nota

Antes de utilizar esta información y el producto al que se refiere, lea la información en “Avisos” en la página 665.

Esta edición se aplica a la versión 9 release 1 de IBM® MQ y a todos los releases y modificaciones posteriores hasta que se indique lo contrario en nuevas ediciones.

Contenido

Protección de IBM MQ

La seguridad es una consideración importante para los desarrolladores de aplicaciones IBM MQ y para los administradores del sistema IBM MQ .

Actualizaciones de seguridad

Asegúrese de que todo el hardware y el software dentro de la zona segura y en las estaciones de trabajo del operador están dentro de su ciclo de vida de soporte, se han actualizado con actualizaciones de software obligatorias y se han aplicado rápidamente las actualizaciones de seguridad.

Puede encontrar más información sobre las actualizaciones de seguridad para:

- Todas las plataformas en IBM Security Bulletins
- APAR de seguridad e integridad del sistema en z/OS en el portal IBM Z System Integrity.

Visión general de la seguridad

Este colección de temas presentan los conceptos de seguridad de IBM MQ.

Primero se presentan los conceptos y mecanismos de seguridad, ya que se aplican a cualquier sistema, seguidos de un debate sobre los mecanismos de seguridad que se han implementado en IBM MQ.

Mecanismos y conceptos de seguridad

Esta colección de temas describe aspectos de la seguridad que se deben tener en cuenta en la instalación de IBM MQ.

Los aspectos de seguridad comúnmente aceptados son los siguientes:

- “Identificación y autenticación” en la página 6
- “Autorización” en la página 6
- “Auditoría” en la página 7
- “Confidencialidad” en la página 7
- “Integridad de datos” en la página 7

Los *mecanismos de seguridad* son herramientas técnicas y métodos técnicos que se utilizan para implementar los servicios de seguridad. Un mecanismo puede funcionar por sí solo, o con otros, para proporcionar un servicio determinado. Los siguientes son ejemplos de mecanismos de seguridad comunes:

- “Criptografía” en la página 7
- “Resúmenes de mensajes y firmas digitales” en la página 9
- “Certificados digitales” en la página 10
- “Infraestructura de claves públicas (PKI)” en la página 14

Cuando planifique una implementación de IBM MQ, considere qué mecanismos de seguridad necesita para implementar estos aspectos de seguridad que son importantes para usted. Para obtener información acerca de lo que ha de tener en cuenta después de que haya leído estos temas, consulte “Planificación de los requisitos de seguridad” en la página 80.

Conceptos relacionados

“Trabajar con SSL/TLS” en la página 277
Estos temas proporcionan instrucciones para realizar tareas individuales relacionados con la utilización de TLS con IBM MQ.

Tareas relacionadas

Conexión de dos gestores de colas utilizando TLS

Identificación y autenticación

La *identificación* es la capacidad de identificar de forma exclusiva a un usuario de un sistema o una aplicación que se está ejecutando en el sistema. La *autenticación* es la capacidad de demostrar que un usuario o una aplicación es realmente quién dicha persona o aplicación asegura ser.

Por ejemplo, considere el caso de un usuario que se conecta a un sistema especificando un ID de usuario y una contraseña. El sistema utiliza el ID de usuario para identificar al usuario. El sistema autentica al usuario en el momento de la conexión comprobando que la contraseña proporcionada es correcta.

No rechazo

El *servicio contra rechazos* se puede considerar una ampliación del servicio de identificación y autenticación. En general, el servicio contra rechazos se aplica cuando se transmiten electrónicamente los datos; por ejemplo, un pedido a un intermediario de bolsa para comprar o vender acciones o una orden de transferencia a un banco de una cuenta a otra.

El objetivo general del servicio contra rechazos es poder demostrar que un mensaje concreto está asociado a un individuo concreto.

El servicio contra rechazos puede contener más de un componente y cada componente proporciona una función diferente. Si el emisor de un mensaje niega alguna vez haberlo enviado, el servicio contra rechazos con *prueba de origen* puede proporcionar al receptor una prueba irrefutable de que el mensaje lo ha enviado esta persona concreta. Si el receptor de un mensaje niega alguna vez haberlo recibido, el servicio contra rechazos con *prueba de entrega* puede proporcionar al emisor una prueba irrefutable de que el mensaje ha sido recibido por esta persona concreta.

En la práctica, obtener una prueba con una seguridad prácticamente del 100% o una prueba irrefutable, es un objetivo difícil de alcanzar. En el mundo real, nada es absolutamente seguro. Gestionar la seguridad está más relacionado con gestionar los riesgos a un nivel que resulte aceptable para la empresa. En este tipo de entornos, la expectativa más realista del servicio contra rechazos es poder proporcionar una prueba que resulte admisible y apoye la causa ante los tribunales.

El servicio contra rechazos es un servicio de seguridad importante en un entorno IBM MQ ya que IBM MQ es un medio de transmitir datos electrónicamente. Por ejemplo, es posible que necesite una prueba actual de que un mensaje determinado lo ha enviado o recibido una aplicación asociada una persona concreta.

IBM MQ con Advanced Message Security no proporciona un servicio contra rechazos como parte de su función básica. No obstante, esta documentación de producto contiene sugerencias sobre cómo puede proporcionar su propio servicio contra rechazos en un entorno IBM MQ escribiendo sus propios programas de salida.

Conceptos relacionados

“Identificación y autenticación en IBM MQ” en la página 21
En IBM MQ, puede implementar la identificación y autenticación utilizando información de contexto de mensaje y autenticación mutua.

Autorización

La *autorización* protege los recursos importantes de un sistema, ya que limita el acceso solamente a los usuarios autorizados y a sus aplicaciones. Impide que los recursos se utilicen sin la autorización necesaria.

Conceptos relacionados

“Autorización en IBM MQ” en la página 22
Puede utilizar la autorización para limitar lo que determinados individuos o aplicaciones pueden hacer en el entorno de IBM MQ.

Auditoría

La *auditoría* es el proceso de registrar y comprobar sucesos para detectar si ha tenido lugar una actividad no esperada o no autorizada, o si se ha llevado a cabo algún intento para realizar dicha actividad.

Para obtener más información acerca de cómo configurar la autorización, consulte "Planificación de la autorización" en la página 84 y los subtemas asociados.

Conceptos relacionados

"Auditoría en IBM MQ" en la página 22
IBM MQ puede emitir mensajes de sucesos para registrar que ha tenido lugar actividad poco usual.

Confidencialidad

El servicio de *confidencialidad* protege la información confidencial para que no pueda divulgarse sin la autorización correspondiente.

Cuando los datos confidenciales se almacenan localmente, los mecanismos de control de accesos pueden ser suficientes para protegerlos basándose en la suposición de que no pueden leerse los datos si no se puede acceder a los mismos. Si se necesita un nivel de seguridad mayor, los datos se pueden cifrar.

Cifre los datos confidenciales cuando se transmitan a través de una red de comunicaciones, especialmente una red insegura, como por ejemplo Internet. En un entorno de red, los mecanismos de control de accesos no son una protección eficaz contra los intentos de interceptar los datos, como por ejemplo, las escuchas telefónicas ilegales.

Integridad de datos

El servicio de *integridad de datos* detecta si se han modificado los datos de forma no autorizada.

Hay dos modos de alterar los datos: de forma accidental, mediante errores de hardware y transmisión o debido a un ataque deliberado. Muchos productos de hardware y protocolos de transmisión disponen de mecanismos para detectar y corregir los errores de hardware y transmisión. La finalidad del servicio de integridad de datos es detectar un ataque deliberado.

El único objetivo del servicio de integridad de datos es detectar si se han modificado los datos. Su objetivo no es restaurar los datos a su estado original si se han modificado.

Los mecanismos de control de accesos pueden ayudar a la integridad de los datos, dado que los datos no se pueden modificar si se deniega el acceso. Pero, del mismo modo que ocurre con la confidencialidad, los mecanismos de control de accesos no resultan eficaces en un entorno de red.

Conceptos de cifrado

En esta colección de temas se describen los conceptos de cifrado aplicables a IBM MQ.

El término *entidad* se utiliza para hacer referencia a un gestor de colas, un IBM MQ MQI client, un usuario individual o cualquier otro sistema capaz de intercambiar mensajes.

Conceptos relacionados

"Cifrado en IBM MQ" en la página 24
IBM MQ proporciona cifrado utilizando el protocolo TLS (seguridad de la capa de transporte).

Criptografía

El cifrado es el proceso de convertir texto legible, denominado *texto plano*, en un formato ilegible, denominado *texto cifrado*.

Esto se produce como se indica a continuación:

1. El emisor convierte el mensaje de texto plano en texto cifrado. Esta parte del proceso se denomina *cifrado* (algunas veces, se denomina *codificación*).
2. El texto cifrado se transmite al receptor.

3. El receptor vuelve a convertir el mensaje de texto cifrado en su formato en texto plano. Esta parte del proceso se denomina *descifrado* (algunas veces, *decodificación*).

La conversión requiere una secuencia de operaciones matemáticas que cambian el aspecto del mensaje durante la transmisión pero no afecta el contenido. Las técnicas de cifrado pueden garantizar la confidencialidad y proteger los mensajes contra la visualización no autorizada (escuchas secretas), ya que un mensaje cifrado no es inteligible. Las firmas digitales, que ofrecen una garantía de la integridad del mensaje, utilizan técnicas de cifrado. Consulte “Firmas digitales en SSL/TLS” en la página 19 para obtener más información.

Las técnicas de cifrado requieren un algoritmo general, que pasa a ser específico mediante el uso de claves. Hay dos clases de algoritmos:

- Los que requieren que ambas partes utilicen la misma clave secreta. Los algoritmos que utilizan una clave compartida se conocen como algoritmos *simétricos*. La Figura 1 en la página 8 ilustra el cifrado de claves simétricas.
- Las que utilizan una clave para cifrado y una clave diferente para descifrado. Una de estas debe mantenerse secreta pero la otra puede ser pública. Los algoritmos que utilizan los pares de claves pública y privada se conocen como algoritmos *simétricos*. La Figura 2 en la página 8 ilustra el cifrado de claves asimétricas, que también se conoce también como *cifrado de claves públicas*.

Los algoritmos de cifrado y descifrado utilizados pueden ser públicos pero la clave secreta compartida y la clave privada debe mantenerse secreta.

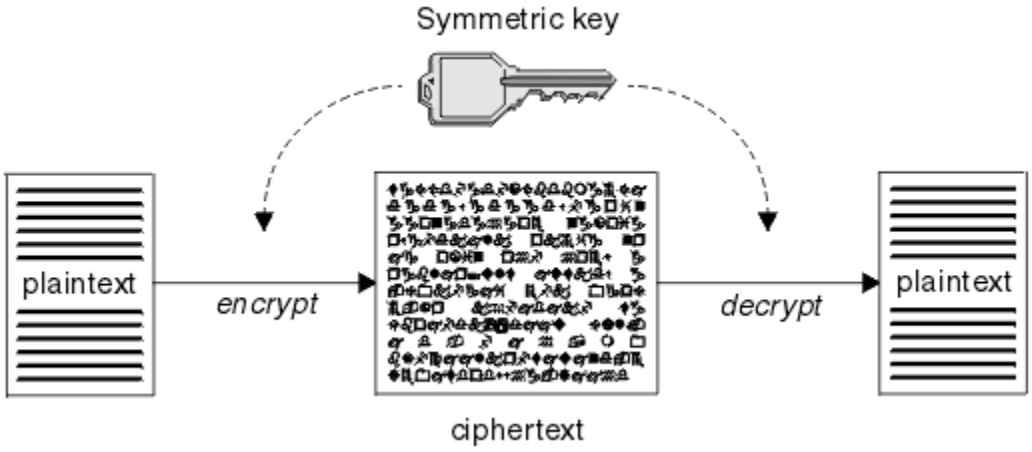


Figura 1. Cifrado de claves simétricas

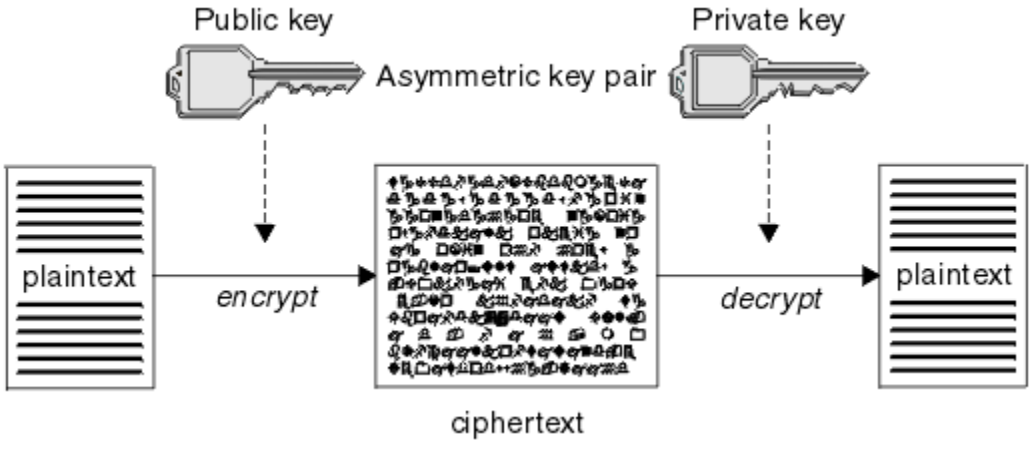


Figura 2. Cifrado de claves asimétricas

La Figura 2 en la página 8 muestra texto plano cifrado con la clave pública del receptor y descifrado con la clave privada del receptor. Solamente el receptor al que va destinado tiene la clave privada para descifrar

el texto cifrado. Tenga en cuenta que el emisor también puede cifrar mensajes con una clave privada, con lo que cualquiera que tenga la clave pública del emisor puede descifrar el mensaje, con la seguridad de que el mensaje procede del emisor.

Con los algoritmos asimétricos, los mensajes se cifran o con la clave pública o con la clave privada pero solamente se pueden descifrar con la otra clave. Solamente la clave privada es secreta, la clave pública la puede conocer cualquiera. Con los algoritmos simétricos, la clave compartida solamente deben conocerla las dos partes. Esto se denomina el *problema de distribución de claves*. Los algoritmos simétricos son más lentos pero tienen la ventaja de que no existe el problema de distribución de claves.

Otra terminología asociada al cifrado es:

Potencia
La potencia del cifrado la determina el tamaño de las claves. Los algoritmos asimétricos requieren claves grandes, por ejemplo:

1024 bits	Clave asimétrica de potencia baja
2048 bits	Clave asimétrica de potencia media
4096 bits	Clave asimétrica de potencia alta

Las claves asimétricas son más pequeñas: las claves de 256 bits le proporcionan un cifrado muy potente.

Algoritmo de cifrado de bloques
Estos algoritmos cifran los datos por bloques. Por ejemplo, el algoritmo RC2 de RSA Data Security Inc. utiliza bloques de 8 bytes de longitud. Los algoritmos de bloques normalmente son más lentos que los algoritmos de flujo.

Algoritmo de cifrado de flujo
Estos algoritmos funcionan en cada byte de datos. Los algoritmos de flujo normalmente son más rápidos que los algoritmos de bloques.

Resúmenes de mensajes y firmas digitales

Un resumen de mensaje es una representación numérica de tamaño fijo del contenido de un mensaje. El resumen del mensaje se calcula mediante una función hash y se puede cifrar, formando una firma digital.

La función hash se utiliza para calcular si un resumen de mensaje cumple con dos criterios:

- Debe ser unidireccional. No debe ser posible invertir la función para encontrar el mensaje correspondiente a un resumen de mensaje específico mediante otros medios que no sea la comprobación de todos los mensajes posibles.
- Debe ser matemáticamente imposible encontrar dos mensajes cuyo valor hash sean iguales al mismo resumen.

El resumen de mensaje se envía con el mensaje propiamente dicho. El receptor puede generar un resumen para el mensaje y compararlo con el resumen del emisor. La integridad del mensaje se verifica cuando los dos resúmenes de mensaje son iguales. Si el mensaje ha sido manipulado de algún modo durante la transmisión, es prácticamente seguro que el resultado sería un resumen de mensaje diferente.

Un resumen de mensaje creado utilizando una clave simétrica secreta es conocido como un Código de Autenticación de Mensaje (MAC), ya que puede garantizar que el mensaje no se ha modificado.

El emisor también puede generar un resumen de mensaje y luego cifrar el resumen utilizando la clave privada de un par de claves asimétricas, formando una firma digital. El receptor debe descifrar luego la firma, antes de compararla con un resumen generado localmente.

Conceptos relacionados

“Firmas digitales en SSL/TLS” en la página 19
Una firma digital se crea cifrando una representación de un mensaje. El cifrado utiliza la clave privada del que firma y, por motivos prácticos, suele operar en un resumen del mensaje en lugar de hacerlo en el mensaje propiamente dicho.

Certificados digitales

Los certificados digitales protegen contra la suplantación de identidad, certificando que una clave pública pertenece a una entidad especificada. Son emitidos por una Entidad emisora de certificados.

Los certificados digitales protegen contra la suplantación de identidad, ya que un certificado digital enlaza una clave pública con su propietario, tanto si el propietario es un usuario, un gestor de colas o cualquier otro tipo de entidad. Los certificados digitales también se denominan certificados de claves públicas ya que le garantizan la propiedad de una clave pública cuando utiliza un esquema de claves asimétrico. Un certificado digital contiene la clave pública para una entidad y es una declaración de que la clave pública pertenece a dicha entidad:

- Cuando el certificado es para una entidad individual, el certificado se denomina *certificado personal* o *certificado de usuario*.
- Cuando el certificado es para una Entidad emisora de certificados, el certificado se denomina *certificado CA* o *certificado de firmante*.

Si las claves públicas las envía directamente su propietario a otra entidad, existe el riesgo de que el mensaje pueda ser interceptado y de que la clave pública sea sustituida por otra. Esto se conoce como *interposición de intrusos*. La solución a este problema es intercambiar las claves públicas mediante una entidad de terceros fiable, con lo que obtiene una mayor garantía de que la clave pública realmente pertenece a la entidad con la que se está comunicando. En lugar de enviar directamente la clave pública, se solicita a la entidad de terceros fiable que la incorpore en un certificado digital. El tercero de confianza que emite certificados digitales se llama autoridad de certificación (CA), tal como se describe en “Entidades emisoras de certificados” en la página 11.

Qué es un certificado digital

Los certificados digitales contienen elementos de información específicos, según se determina en el estándar X.509.

Los certificados digitales que utiliza IBM MQ se ajustan al estándar X.509, que especifica la información necesaria y el formato con que se envía. X.509 es la parte de la infraestructura de autenticación de las series de estándares X.500.

Los certificados digitales contienen como mínimo la información siguiente acerca de la entidad que se está certificando:

- La clave pública del propietario
- El nombre distinguido del propietario
- El Nombre distinguido de la CA que ha emitido el certificado
- La fecha a partir de la cual es válido el certificado
- La fecha de caducidad del certificado
- El número de versión del formato de datos del certificado como se define en x.509. La versión actual del estándar x.509 es la Versión 3, y la mayoría de los certificados se ajustan a dicha versión.
- Un número de serie. Se trata de un identificador exclusivo asignado por la CA que emitió el certificado. El número de serie es exclusivo dentro de la CA que emitió el certificado: no hay dos certificados firmados por el mismo certificado de CA que tengan el mismo número de serie.

Un certificado X.509 Versión 2 también contiene un Identificador de emisor y un Identificador de sujeto, y un certificado X.509 Versión 3 puede contener varias extensiones. Algunas extensiones de certificados, como por ejemplo la extensión de restricción básica, son *estándar* pero otras son específicas de la implementación. Una extensión puede ser *crítica*, en cuyo caso debe haber un sistema disponible para reconocer el campo; si no reconoce el campo, deberá rechazar el certificado. Si una extensión no es crítica, el sistema podrá omitirla si no la reconoce.

La firma digital de un certificado personal se genera utilizando la clave privada de la CA que ha firmado dicho certificado. Cualquier persona que necesite verificar el certificado personal puede utilizar la clave pública de la CA para hacerlo. El certificado de la CA contiene la clave pública.

Los certificados digitales no contienen la clave privada. Debe mantener la clave privada en secreto.

Requisitos para los certificados personales
IBM MQ da soporte a certificados digitales que cumplan con el estándar X.509. Requiere la opción de autenticación de cliente.

Dado que IBM MQ es un sistema de igual a igual, se considera como una autenticación de cliente en la terminología de SSL/TLS. Por consiguiente, cualquier certificado personal utilizado para la autenticación SSL/TLS debe permitir un uso de claves de autenticación de cliente. No todos los certificados de servidor tienen esta opción habilitada, por lo que es posible que el proveedor de certificados tenga que habilitar la autenticación de cliente en la CA raíz para un certificado seguro.

Además de los estándares que especifican el formato de datos para un certificado digital, también existen los estándares para determinar si un certificado es válido. Estos estándares se han actualizado a lo largo del tiempo para impedir ciertos tipos de infracción de seguridad. Por ejemplo, los certificados X.509 anteriores de la versión 1 y 2 no indicaban si el certificado se podía utilizar legítimamente para firmar otros certificados. Por lo tanto, era posible que un usuario malicioso obtuviese un certificado personal de un origen legítimo y crease nuevos certificados diseñados para suplantar a otros usuarios.

Al utilizar certificados X.509 de la versión 3, las extensiones de certificado BasicConstraints y KeyUsage se utilizan para especificar qué certificados pueden firmar legítimamente otros certificados. El estándar IETF RFC 5280 especifica una serie de reglas de validación de certificados que el software de aplicación compatible debe implementar para evitar ataques de suplantación. Un conjunto de reglas de certificado se conoce como una política de validación de certificados.

Para obtener más información sobre las políticas de validación de certificados en IBM MQ, consulte "Políticas de validación de certificados en IBM MQ" en la página 44.

Entidades emisoras de certificados
Una Entidad emisora de certificados (CA) es una entidad de terceros fiable que emite certificados digitales que le garantizan que la clave pública de una entidad pertenece realmente a dicha entidad.

Las funciones de una CA son:

- Al recibir una solicitud de un certificado digital, verificar la identidad del solicitante antes de crear, firmar y devolver el certificado personal.
- Proporcionar la clave pública propia de la CA en su certificado de CA.
- Publicar listas de certificados que ya no son fiables en la Lista de revocación de certificados (CRL). Para obtener más información, consulte "Trabajar con certificados revocados" en la página 342.
- Proporcionar acceso al estado de revocación del certificado utilizando un servidor de programa de respuesta OCSP

Nombres distinguidos
El nombre distinguido (DN) identifica de forma exclusiva una entidad en un certificado X.509.

Atención: En un filtro SSLPEER solo pueden utilizarse los atributos de la tabla siguiente. Los nombres distinguidos de certificado pueden contener otros atributos, pero no se permite filtrar en estos atributos.

Tabla 1. Los tipos de atributo encontrados en el nombre distinguido que se pueden utilizar en un filtro SSLPEER

Tipo de atributo	**Descripción**
SERIALNUMBER	Número de serie de certificado
MAIL	Dirección de correo electrónico
E	Dirección de correo electrónico (En desuso por ser preferible MAIL)
UID o USERID	Identificador de usuario
CN	Nombre común

Tabla 1. Los tipos de atributo encontrados en el nombre distinguido que se pueden utilizar en un filtro SSLPEER (continuación)	
Tipo de atributo	**Descripción**
T	Título
OU	Nombre de la unidad organizativa
DC	Componente de dominio
O	Nombre de la organización
CALLE	Calle / Primera línea de dirección
L	Nombre de la localidad
ST (o SP o S)	Nombre del estado o provincia
PC	Código postal
C	País
UNSTRUCTUREDNAME	Nombre de host
UNSTRUCTUREDADDRESS	Dirección IP
DNQ	Calificador de nombre distinguido

El estándar X.509 define otros atributos que generalmente no forman parte del DN pero que pueden proporcionar extensiones opcionales al certificado digital.

El estándar X.509 proporciona un DN que se especifica con un formato de serie. Por ejemplo:

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

El Nombre común (CN) puede describir un usuario individual o cualquier otra entidad, por ejemplo un servidor web.

El DN puede contener varios atributos OU y DC. Sólo se permite una instancia de cada uno de los otros atributos. El orden de las entradas OU es importante: el orden especifica una jerarquía de nombres de unidades organizativas, con el nivel de unidad del más alto nivel en primer lugar. El orden de las entradas DC también es importante.

IBM MQ tolera ciertos nombres distinguidos (DN) malformados. Para obtener más información, consulte Reglas de IBM MQ para valores de SSLPEER.

Conceptos relacionados

“Qué es un certificado digital” en la página 10
Los certificados digitales contienen elementos de información específicos, según se determina en el estándar X.509.

Obtención de certificados personales de una entidad emisora de certificados
Puede obtener un certificado de una entidad emisora de certificados (CA) externa.

Un certificado digital se obtiene enviando información a un CA, en forma de una solicitud de certificado. El estándar X.509 define un formato para esta información, pero algunas CA tienen su propio formato. Las solicitudes de certificados las generan normalmente la herramienta de gestión de certificados que el sistema utiliza; por ejemplo:

- Multi La herramienta iKeyman en Multiplatforms.
- z/OS RACF en z/OS.

La información contiene el Nombre distinguido y la clave pública. Cuando la herramienta de gestión de certificados genera la solicitud de certificado, también genera la clave privada, que debe mantener en un lugar seguro. No distribuya nunca su clave privada.

Cuando la CA recibe la solicitud, la autorización comprueba su identidad antes de crear el certificado y devolverlo como un certificado personal.

La Figura 3 en la página 13 ilustra el proceso de obtener un certificado digital de una CA.

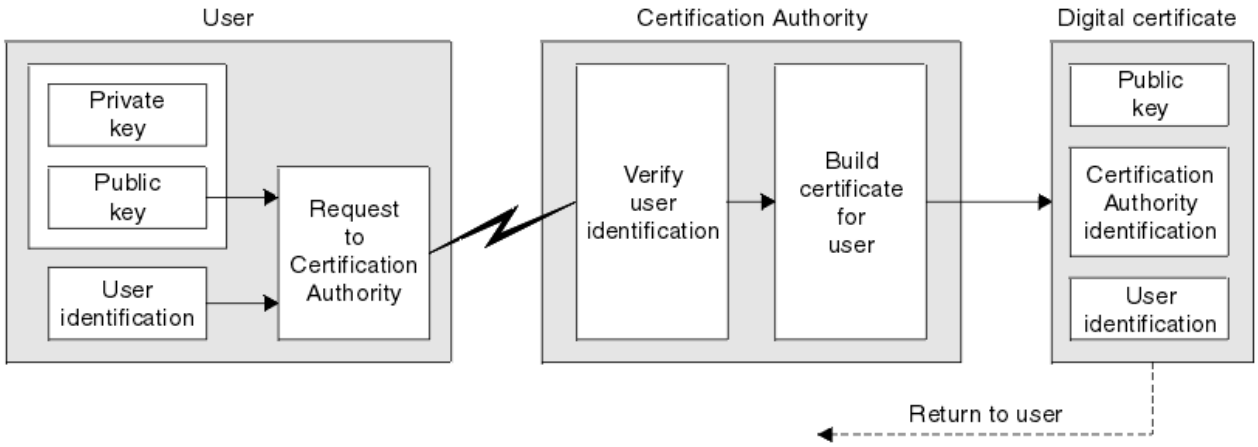


Figura 3. Obtención de un certificado digital

En el diagrama:

- La identificación de usuario incluye su Nombre distinguido de sujeto.
- La identificación de autoridad de certificación incluye el Nombre distinguido de la CA que está emitiendo el certificado.

Los certificados digitales contienen campos adicionales distintas de las que aparecen en el diagrama. Para obtener más información sobre el resto de campos en un certificado digital, consulte “Qué es un certificado digital” en la página 10.

Cómo funcionan las cadenas de certificados
Cuando recibe el certificado de otra entidad, es posible que necesite utilizar una *cadena de certificados* para obtener el certificado de la *CA raíz*.

La cadena de certificados, que también se conoce como la *vía de acceso de certificación*, es una lista de certificados que se utiliza para autenticar una entidad. La cadena, o vía de acceso, comienza por el certificado de esta entidad y cada uno de los certificados de la cadena lo forma la entidad identificada mediante el certificado siguiente de la cadena. La cadena termina con un certificado de la CA raíz. El certificado de la CA raíz siempre está firmado por la propia entidad emisora de certificados (CA). Las firmas de todos los certificados de la cadena se deben verificar hasta que se alcance el certificado de CA raíz.

La Figura 4 en la página 14 ilustra una vía de acceso de certificación desde el propietario del certificado a la CA raíz, donde la cadena de confianza comienza.

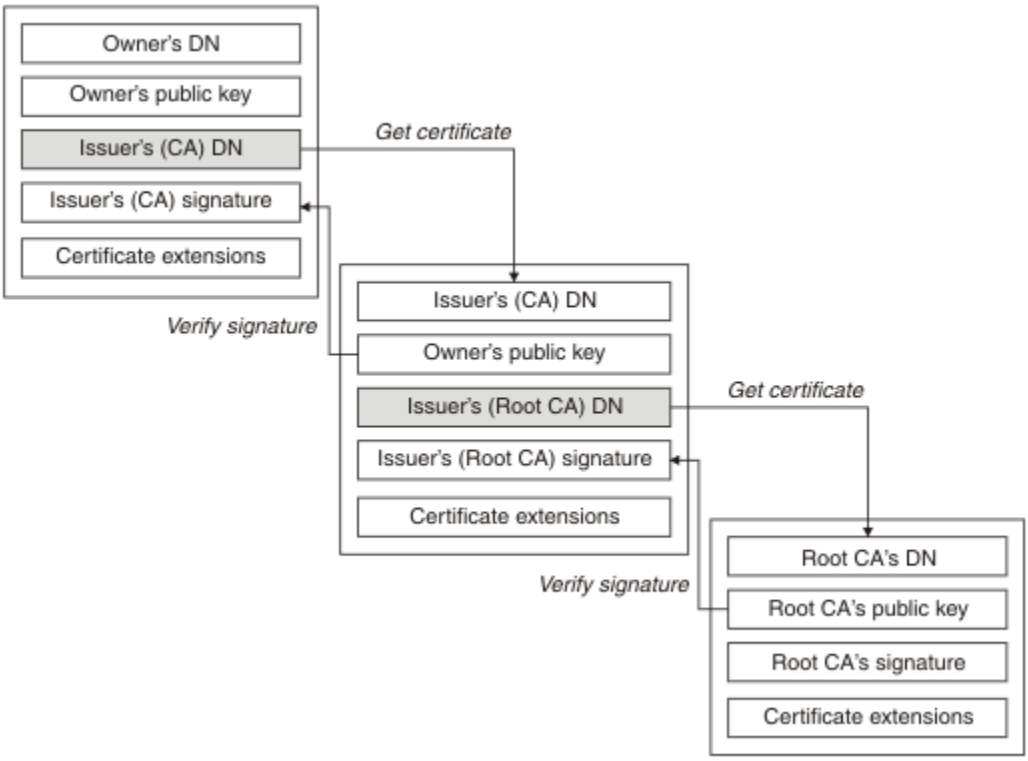


Figura 4. Cadena fiable

Cada certificado puede contener una o varias extensiones. Un certificado que pertenece a una CA contiene normalmente una extensión BasicConstraints con el distintivo isCA establecido para indicar que está permitido firmar otros certificados.

Cuando los certificados ya no son válidos
Los certificados digitales pueden caducar o revocarse.

Los certificados digitales se emiten durante un período fijo de tiempo y no son válidos después de su fecha de caducidad.

Los certificados se pueden revocar por varios motivos, entre ellos:

- El propietario ha cambiado a una organización distinta.
- La clave privada ya no es secreta.

IBM MQ puede comprobar si un certificado se ha revocado enviando una solicitud a un respondedor OCSP (Online Certificate Status Protocol) (en UNIX, Linux®, and Windows solamente). Alternativamente, pueden acceder a una CRL (Certificate Revocation List) en un servidor LDAP. La información de revocación OCSP y de CRL la publica una entidad emisora de certificados. Para obtener más información, consulte "Trabajar con certificados revocados" en la página 342.

Infraestructura de claves públicas (PKI)

Una infraestructura de claves públicas (PKI) es un sistema de recursos, políticas y servicios que da soporte al uso del cifrado de claves públicas para autenticar a las partes que participan en una transacción.

No hay ningún estándar individual que defina los componentes de una Infraestructura de clave pública, pero normalmente un PKI consta de entidades emisoras de certificados (CA) y entidades emisoras de registro (RA). Las CA proporcionan los servicios siguientes:

- Emisión de certificados digitales
- Validación de certificados digitales
- Revocación de certificados digitales

- Distribución de claves públicas

El estándar X.509 proporcionan la base para la industria estándar de la infraestructura Public Key Infrastructure.

Consulte “Certificados digitales” en la página 10 para obtener más información sobre los certificados digitales y las entidades emisoras de certificados (CA). Las RA verifican la información que se proporciona cuando se solicitan certificados digitales. Si la RA verifica esta información, la CA puede emitir un certificado digital para el solicitante.

Una PKI también puede proporcionar las herramientas para gestionar los certificados digitales y las claves públicas. Una PKI se describe a veces como una *jerarquía fiable* para la gestión de certificados digitales, aunque la mayor parte de las definiciones incluyen servicios adicionales. Algunas definiciones incluyen servicios de cifrado y firma digital, pero estos servicios no son esenciales para el funcionamiento de una PKI.

Protocolos de seguridad de cifrado: TLS

Los protocolos de cifrado proporcionan conexiones seguras, permitiendo que dos partes se comuniquen con privacidad e integridad de datos. El protocolo TLS (Transport Layer Security) ha evolucionado a partir del protocolo SSL (Secure Sockets Layer). IBM MQ ofrece soporte para TLS.

Los principales objetivos de ambos protocolos consisten en proporcionar confidencialidad, (que a veces recibe el nombre de *privacidad*), integridad de datos, identificación y autenticación utilizando certificados digitales.

Aunque los dos protocolos son parecidos, las diferencias son suficientemente significativas como para que SSL 3.0 y las diversas versiones de TLS no puedan interactuar.

Conceptos relacionados

“Protocolos de seguridad TLS en IBM MQ” en la página 24
IBM MQ da soporte al protocolo TLS (seguridad de la capa de transporte) para proporcionar seguridad a nivel de enlace para los canales de mensajes y los canales MQI.

Conceptos de TLS (Transport Layer Security)

El protocolo TLS permite que dos partes se identifiquen y autentiquen entre sí y se comuniquen con confidencialidad e integridad de datos. El protocolo TLS ha evolucionado a partir del protocolo Netscape SSL 3.0, pero TLS y SSL no pueden interactuar.

El protocolo TLS proporciona a las comunicaciones seguridad en Internet y permiten a las aplicaciones cliente/servidor comunicarse de una forma que es confidencial y fiable. Los protocolos tienen dos capas: un protocolo de registro y un protocolo de reconocimiento y éstos están en capas por encima de un protocolo de transporte como, por ejemplo, TCP/IP. Ambos utilizan técnicas de cifrado simétrico y asimétrico.

Una aplicación inicia una conexión TLS, que se convierte en el cliente TLS. La aplicación que recibe la conexión pasa a ser el servidor TLS. Cada nueva sesión se inicia con un reconocimiento, tal como lo define el protocolo TLS.

Se proporciona una lista completa de las CipherSpecs soportadas por IBM MQ en “Habilitación de CipherSpecs” en la página 426.

Para obtener información sobre el protocolo SSL, consulte la información que se proporciona en https://www.mozilla.org/projects/security/pki/nss/ssl/draft302.txt. Para obtener información sobre el protocolo TLS, consulte la información proporcionada por TLS Working Group en el sitio web de Internet Engineering Task Force en https://www.ietf.org

Visión general del reconocimiento SSL/TLS

El reconocimiento SSL/TLS permite que el cliente y el servidor TLS establezcan las claves secretas con las que se comunican.

Esta sección proporciona un resumen de los pasos que permiten que el cliente y el servidor TLS se comuniquen entre sí.

- Acordar la versión del protocolo que se va a utilizar.
- Seleccionar los algoritmos de cifrado.
- Autenticarse mutuamente intercambiando y validando certificados digitales.
- Utilizar técnicas de cifrado asimétrico para generar una clave secreta compartida, que evita el problema de distribución de claves. TLS utiliza entonces la clave compartida para el cifrado simétrico de los mensajes, lo cual es más rápido que el cifrado asimétrico.

Para obtener más información acerca de los algoritmos de cifrado y los certificados digitales, consulte la información relacionada.

En general, los pasos que se realizan durante el reconocimiento TLS son los siguientes:

1. El cliente TLS envía un mensaje de "saludo del cliente" que lista la información de cifrado como, por ejemplo, la versión de TLS y, según el orden de preferencias del cliente, las CipherSuites que soporta el cliente. El mensaje también contiene un serie de bytes aleatorios que se utilizan en cálculos posteriores. El protocolo permite que el mensaje de "saludo del cliente" incluya los métodos de compresión de datos soportados por el cliente.
2. El servidor TLS responde con un mensaje de "saludo del servidor" que contiene la CipherSuite elegida por el servidor en la lista que ha proporcionado el cliente, el ID de sesión y otra serie de bytes aleatorios. El servidor también envía su certificado digital. Si el servidor requiere un certificado digital para la autenticación del cliente, el servidor envía una "solicitud de certificado de cliente" que incluye una lista de los tipos de certificados soportados y los nombres distinguidos de las Autoridades de certificación (CA) aceptables.
3. El cliente TLS verifica el certificado digital del servidor. Para obtener más información, consulte “Cómo proporciona TLS identificación, autenticación, confidencialidad e integridad” en la página 17.
4. El cliente TLS envía la serie de bytes aleatorios que permite que tanto el cliente como el servidor calculen la clave secreta que se utilizará para cifrar los datos del mensaje posterior. La serie de bytes aleatorios se cifra con la clave pública del servidor.
5. Si el servidor TLS ha enviado una "solicitud de certificado de cliente", el cliente envía una serie de bytes aleatorios cifrada con la clave privada del cliente, junto con el certificado digital del cliente, o una "alerta que indica que no hay certificado digital". Esta alerta es simplemente un aviso, pero en algunas implementaciones el reconocimiento no se ejecuta correctamente si la autenticación de cliente es obligatoria.
6. El servidor TLS verifica el certificado del cliente. Para obtener más información, consulte “Cómo proporciona TLS identificación, autenticación, confidencialidad e integridad” en la página 17.
7. El cliente TLS envía al servidor un mensaje de "finalizado", que se cifra con la clave secreta, que indica que la parte de cliente del reconocimiento se ha completado.
8. El servidor TLS envía al cliente un mensaje de "finalizado", que se cifra con la clave secreta, que indica que la parte de servidor del reconocimiento se ha completado.
9. Durante la sesión TLS, el servidor y el cliente podrán intercambiar mensajes que estén cifrados simétricamente con la clave secreta compartida.

La Figura 5 en la página 17 ilustra el reconocimiento TLS.

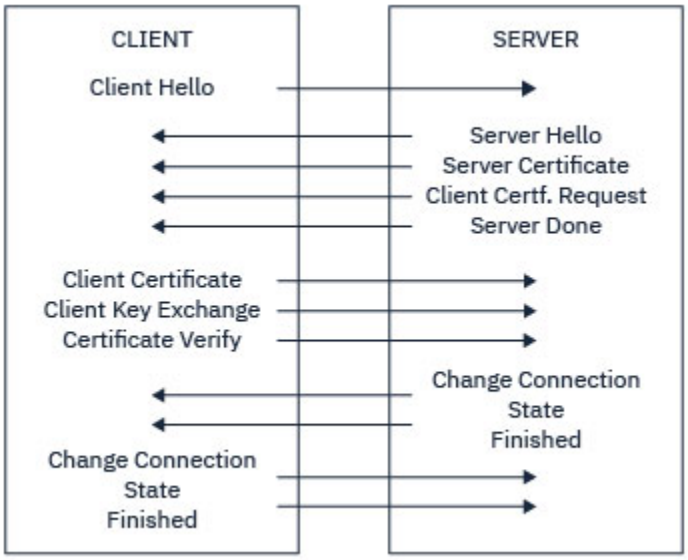


Figura 5. Visión general del reconocimiento TLS

Cómo proporciona TLS identificación, autenticación, confidencialidad e integridad

Durante la autenticación del cliente y servidor hay un paso que requiere que se cifren los datos con una de las claves de un par de claves asimétricas y que se descifren con la otra clave del par. Se utiliza un resumen de mensaje para proporcionar integridad.

Para obtener una visión general de los pasos implicados en el reconocimiento TLS, consulte "Visión general del reconocimiento SSL/TLS" en la página 15.

Cómo proporciona TLS autenticación

Para la autenticación del servidor, el cliente utiliza la clave pública del servidor para cifrar los datos que ha utilizado para calcular la clave secreta. El servidor puede generar la clave secreta solamente si puede descifrar los datos con la clave privada correcta.

Para la autenticación del cliente, el servidor utiliza la clave pública del certificado de cliente para descifrar los datos que el cliente envía durante el paso "5" en la página 16 del reconocimiento. El intercambio de mensajes cifrados con las claves secretas que indican que ha finalizado (los pasos "7" en la página 16 y "8" en la página 16 de la visión general) confirma que se ha completado la autenticación.

Si cualquiera de los pasos de autenticación falla, el reconocimiento no se ejecutará correctamente y la sesión finalizará.

El intercambio de certificados digitales durante el reconocimiento TLS forma parte del proceso de autenticación. Para obtener más información acerca de cómo los certificados ofrecen protección contra la suplantación de identidad, consulte la información relacionada. Los certificados necesarios son los siguientes, siendo la CA X la que emite el certificado para el cliente TLS y la CA Y la que emite el certificado para el servidor TLS:

Sólo para la autenticación del servidor, el servidor TLS necesita:

- El certificado personal que la CA Y ha emitido para el servidor
- La clave privada del servidor

y el cliente TLS necesita:

- El certificado de CA de la CA Y

Si el servidor TLS requiere autenticación de cliente, el servidor verifica la identidad del cliente verificando el certificado digital del cliente con la clave pública para la CA que ha emitido el certificado personal al cliente, en este caso CA X. Para la autenticación del servidor y del cliente, el servidor necesita:

- El certificado personal que la CA Y ha emitido para el servidor
- La clave privada del servidor
- El certificado de CA de la CA X

y el cliente necesita:

- El certificado personal que la CA X ha emitido para el cliente
- La clave privada del cliente
- El certificado de CA de la CA Y

Es posible que tanto el servidor como el cliente TLS necesiten otros certificados de CA para formar una cadena de certificados hasta el certificado de CA raíz. Para obtener más información acerca de las cadenas de certificados, consulte la información relacionada.

Qué ocurre durante la verificación de certificados

Como se ha indicado en los pasos "3" en la página 16 y "6" en la página 16 de la visión general, el cliente TLS verifica el certificado del servidor, y el servidor TLS verifica el certificado del cliente. Hay cuatro aspectos en esta verificación:

1. La firma digital se comprueba (consulte "Firmas digitales en SSL/TLS" en la página 19).
2. La cadena de certificados se comprueba; también debe tener certificados de CA intermedios (consulte "Cómo funcionan las cadenas de certificados" en la página 13).
3. Las fechas de activación y caducidad y el período de validez se comprueban.
4. Se comprueba el estado de revocación del certificado (consulte "Trabajar con certificados revocados" en la página 342).

Restablecimiento de claves secretas

Durante el reconocimiento TLS se genera una *clave secreta* que sirve para cifrar los datos que se transfieren del cliente al servidor TLS. La clave secreta utiliza una fórmula matemática que se aplica a los datos para transformar texto plano en texto cifrado ilegible y, texto cifrado en texto plano.

La clave secreta se genera a partir de un texto aleatorio que se envía como parte del reconocimiento y se utiliza para convertir texto plano en texto cifrado. La clave secreta también se utiliza en el algoritmo MAC (Código de autenticación de mensaje), que se utiliza para determinar si se ha modificado un mensaje. Consulte "Resúmenes de mensajes y firmas digitales" en la página 9 para obtener más información.

Si se descubre la clave secreta, podría descifrarse el texto plano de un mensaje a partir del texto cifrado o podría calcularse un resumen del mensaje, que permitiría alterar mensajes sin detectarlo. Incluso para un algoritmo complejo podría llegar a descubrirse el texto plano aplicando cada una de las transformaciones matemáticas posibles al texto cifrado. Para reducir la cantidad de datos que puede descifrarse o modificarse si se descubre la clave secreta, la clave puede negociarse de nuevo periódicamente. Cuando se ha negociado la clave secreta, la clave secreta anterior ya no se podrá utilizar para descifrar datos cifrados con la nueva clave secreta.

Cómo proporciona TLS confidencialidad

TLS utiliza una combinación de cifrado simétrico y asimétrico para garantizar la privacidad de mensajes. Durante el reconocimiento TLS, el cliente y el servidor TLS acuerdan el uso de un algoritmo de cifrado y de una clave secreta compartida que se emplearán sólo para una sesión. Todos los mensajes transmitidos entre el cliente y el servidor TLS se cifran utilizando este algoritmo y esta clave, lo que garantiza la confidencialidad del mensaje incluso si resulta interceptado. Dado que TLS utiliza cifrado asimétrico durante el transporte de la clave secreta compartida, no hay ningún problema de distribución de claves. Para obtener más información acerca de las técnicas de cifrado, consulte "Criptografía" en la página 7.

Cómo proporciona TLS integridad

TLS proporciona integridad de datos calculando un resumen del mensaje. Para obtener más información, consulte “Integridad de datos de mensajes” en la página 464.

El uso de TLS garantiza la integridad de los datos, siempre que la CipherSpec en la definición de canal utilice un algoritmo de hash tal como se describe en la tabla en “Habilitación de CipherSpecs” en la página 426.

En concreto, si la integridad de los datos es una preocupación, debe evitar elegir un CipherSpec cuyo algoritmo hash se muestre como "None". El uso de MD5 también está muy desaconsejado, ya que ahora es muy antiguo y ya no es seguro para la mayoría de los objetivos prácticos.

CipherSpecs y CipherSuites

Los protocolos de seguridad criptográficos deben estar de acuerdo con los algoritmos utilizados por una conexión segura. CipherSpecs y CipherSuites definen combinaciones específicas de algoritmos.

Una CipherSpec identifica una combinación de algoritmo de cifrado y algoritmo MAC (Message Authentication Code). Ambos extremos de una conexión TLS deben estar de acuerdo en la misma CipherSpec para poder comunicarse.

IBM MQ admite el protocolo TLS 1.2. Sin embargo, puede habilitar las CipherSpecs en desuso, si tiene que hacerlo.

Consulte “Habilitación de CipherSpecs” en la página 426 si desea información sobre:

- CipherSpecs soportadas por IBM MQ
- Cómo se habilitan las CipherSpecs SSL 3.0 y TLS 1.0 en desuso

Importante: Cuando se trata con canales IBM MQ, se debe utilizar una CipherSpec. Cuando se trata con canales Java, canales JMS o canales MQTT debe especificar una CipherSuite.

Para obtener más información sobre CipherSpecs, consulte “Habilitación de CipherSpecs” en la página 426.

Una CipherSuite es una suite de algoritmos de cifrado que utiliza una conexión TLS. Una suite consta de tres algoritmos diferentes:

- El algoritmo de intercambio y autenticación de claves, que se utiliza durante el reconocimiento SSL
- El algoritmo de cifrado, que se utiliza para cifrar los datos
- El algoritmo MAC (Código de autenticación de mensaje), que se utiliza para generar el resumen de mensaje

Hay varias opciones para cada componente de la suite, pero solo ciertas combinaciones son válidas cuando se especifican para una conexión TLS. El nombre de una CipherSuite válida define la combinación de algoritmos utilizados. Por ejemplo, la CipherSuite TLS_RSA_WITH_AES_128_CBC_SHA especifica:

- El algoritmo de intercambio y autenticación de claves RSA
- El algoritmo de cifrado AES, utilizando una clave de 128 bits y una modalidad de encadenamiento de bloques de cifrado (CBC)
- El código de autenticación de mensaje SHA-1 (MAC)

Firmas digitales en SSL/TLS

Una firma digital se crea cifrando una representación de un mensaje. El cifrado utiliza la clave privada del que firma y, por motivos prácticos, suele operar en un resumen del mensaje en lugar de hacerlo en el mensaje propiamente dicho.

Las firmas digitales varían según los datos que se firman, a diferencia de las firmas manuales, que no dependen del contenido del documento que se firma. Si la misma entidad firma digitalmente dos mensajes diferentes, las dos firmas serán diferentes pero ambas pueden verificarse con la misma clave pública, es decir, la clave pública de la entidad que ha firmado los mensajes.

Los pasos del proceso de firma digital son los siguientes:

1. El emisor calcula un resumen de un mensaje y, a continuación, cifra el resumen utilizando la clave privada del emisor, para formar la firma digital.
2. El emisor transmite la firma digital con el mensaje.
3. El receptor descifra la firma digital utilizando la clave pública del emisor y vuelve a generar el resumen del mensaje del emisor.
4. El receptor calcula un resumen del mensaje a partir de los datos del mensaje que recibe y comprueba que los dos resúmenes sean iguales.

La Figura 6 en la página 20 ilustra este proceso.

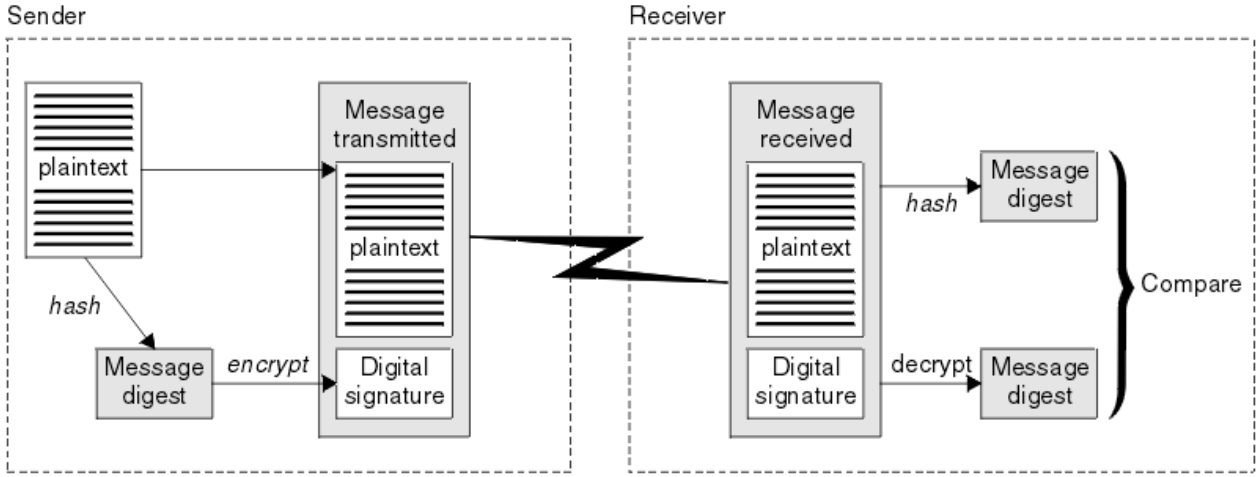


Figura 6. El proceso de firma digital

Si se verifican las dos firmas digitales, el receptor sabe que:

- El mensaje no se ha modificado durante la transmisión.
- El mensaje lo ha enviado la entidad que asegura haberlo enviado.

Las firmas digitales forman parte de los servicios de integridad y autenticación. Las firmas digitales también proporcionan una prueba de origen. Solamente el emisor conoce la clave privada, que proporciona una prueba irrefutable de que el emisor es quien ha originado el mensaje.

Nota: También puede descifrar el mensaje propiamente dicho, lo cual protege la confidencialidad de la información que contiene el mensaje.

Federal Information Processing Standards

El gobierno de EE.UU. ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. El NIST (National Institute for Standards and Technology) es un importante ente al que le preocupan las cuestiones relativas a los sistemas de TI y la seguridad. El NIST propone recomendaciones y genera estándares, entre los que se incluye FIPS (Federal Information Processing Standards).

Uno de estos significativos estándares es FIPS 140-2, que requiere el uso de algoritmos de cifrado fuerte. FIPS 140-2 también especifica los requisitos para que algoritmos de hash se puedan utilizar para proteger los paquetes contra su modificación mientras están en tránsito.

IBM MQ proporciona soporte para FIPS 140-2 cuando se ha configurado para a tal efecto.

Con el tiempo, los analistas desarrollan ataques contra los algoritmos de cifrado y de hash existentes. Se adoptan nuevos algoritmos para poder resistir dichos ataques. FIPS 140-2 se actualiza periódicamente para tener en cuenta estos cambios.

Conceptos relacionados

“Cifrado Suite B de la NSA (National Security Agency)” en la página 21

El gobierno de los Estados Unidos de América ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. La NSA (National Security Agency) de Estados Unidos recomienda un conjunto de algoritmos de cifrado interoperables en su estándar Suite B.

Cifrado Suite B de la NSA (National Security Agency)

El gobierno de los Estados Unidos de América ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. La NSA (National Security Agency) de Estados Unidos recomienda un conjunto de algoritmos de cifrado interoperables en su estándar Suite B.

El estándar Suite B especifica una modalidad de funcionamiento en la que sólo se utiliza un conjunto específico de algoritmos de cifrado. El estándar Suite B especifica lo siguiente:

- El algoritmo de cifrado (AES)
- El algoritmo de intercambio de claves (Elliptic Curve Diffie-Hellman, también conocido como ECDH)
- El algoritmo de firma digital (Elliptic Curve Digital Signature Algorithm, también conocido como ECDSA)
- Los algoritmos de hash (SHA-256 o SHA-384)

Además, el estándar IETF RFC 6460 especifica perfiles compatibles con Suite B que definen la configuración de la aplicación y el comportamiento detallados necesarios para cumplir estándar Suite B. Define dos perfiles:

1. Un perfil compatible con Suite B que puede utilizarse con TLS 1.2. Cuando se configure para el funcionamiento compatible con Suite B, sólo se utilizará el conjunto restringido de algoritmos de cifrado enumerados.
2. Un perfil transitorio para su uso con TLS 1.0 o TLS 1.1. Este perfil permite la interoperatividad con servidores que no sean compatibles con Suite B. Cuando se configura para el funcionamiento transitorio de Suite B, pueden utilizarse algoritmos de cifrado y de hash adicionales.

El estándar Suite B es conceptualmente parecido a FIPS 140-2, porque restringe el conjunto de algoritmos de cifrado permitidos para proporcionar un nivel de seguridad garantizado.

En sistemas Windows, UNIX and Linux, IBM MQ, puede configurarse para que se ajuste el perfil TLS 1.2 compatible con Suite B, pero no da soporte al perfil transitorio de Suite B. Para obtener más información, consulte “NSA Suite B Cryptography en IBM MQ” en la página 41.

Referencia relacionada

“Federal Information Processing Standards” en la página 20
El gobierno de EE.UU. ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. El NIST (National Institute for Standards and Technology) es un importante ente al que le preocupan las cuestiones relativas a los sistemas de TI y la seguridad. El NIST propone recomendaciones y genera estándares, entre los que se incluye FIPS (Federal Information Processing Standards).

Mecanismos de seguridad de IBM MQ

En esta colección de temas se explica cómo se pueden implementar los distintos conceptos de seguridad en IBM MQ.

IBM MQ proporciona mecanismos para implementar todos los conceptos de seguridad introducidos en “Mecanismos y conceptos de seguridad” en la página 5. Estos se describen más detalladamente en las siguientes secciones.

Identificación y autenticación en IBM MQ

En IBM MQ, puede implementar la identificación y autenticación utilizando información de contexto de mensaje y autenticación mutua.

A continuación se muestran algunos ejemplos de la identificación y autenticación en un entorno de IBM MQ:

- Todo mensaje puede contener información de *contexto de mensaje*. Esta información se guarda en el descriptor de mensaje. La puede generar el gestor de colas cuando una aplicación transfiere un mensaje a una cola. Alternativamente, la aplicación puede proporcionar la información si el ID de usuario asociado a la aplicación tiene autorización para hacerlo.

  La información de contexto que contiene un mensaje permite que la aplicación receptora obtenga información acerca del emisor del mensaje. Por ejemplo, contiene el nombre de la aplicación que ha transferido el mensaje y el ID de usuario asociado a la aplicación.
- Cuando se inicia un canal de mensajes, el agente de canal de mensajes (MCA) de cada extremo del canal puede autenticar a su asociado. Esta técnica se conoce como *autenticación mutua*. Para el MCA emisor, ofrece la garantía de que el asociado que está a punto de enviar los mensajes es auténtico. Para el MCA receptor, hay una garantía similar de que está a punto de recibir mensajes de un asociado auténtico.

Conceptos relacionados

"Identificación y autenticación" en la página 6
La *identificación* es la capacidad de identificar de forma exclusiva a un usuario de un sistema o una aplicación que se está ejecutando en el sistema. La *autenticación* es la capacidad de demostrar que un usuario o una aplicación es realmente quién dicha persona o aplicación asegura ser.

Autorización en IBM MQ

Puede utilizar la autorización para limitar lo que determinados individuos o aplicaciones pueden hacer en el entorno de IBM MQ.

A continuación se muestran algunos ejemplos de autorización en un entorno de IBM MQ:

- Permitir solamente que el administrador autorizado pueda emitir mandatos para gestionar recursos de IBM MQ.
- Permitir que una aplicación se conecte a un gestor de colas solamente si el ID de usuario asociado a la aplicación tiene autorización para hacerlo.
- Permitir que una aplicación abra solamente las colas que sean necesarias para su funcionamiento.
- Permitir que una aplicación se suscriba solamente a los temas que sean necesarios para su funcionamiento.
- Permitir que una aplicación realice en una cola solamente las operaciones que sean necesarias para su funcionamiento. Por ejemplo, es posible que una aplicación sólo necesite examinar los mensajes de una cola determinada y no necesite transferir ni obtener mensajes.

Para obtener más información acerca de cómo configurar la autorización, consulte "Planificación de la autorización" en la página 84 y los subtemas asociados.

Conceptos relacionados

"Autorización" en la página 6
La *autorización* protege los recursos importantes de un sistema, ya que limita el acceso solamente a los usuarios autorizados y a sus aplicaciones. Impide que los recursos se utilicen sin la autorización necesaria.

Auditoría en IBM MQ

IBM MQ puede emitir mensajes de sucesos para registrar que ha tenido lugar actividad poco usual.

A continuación se muestran algunos ejemplos de auditoría en un entorno de IBM MQ:

- Una aplicación intenta abrir una cola que no tiene autorización para abrir. Se emite un mensaje de suceso de instrumentación. Al inspeccionar el mensaje de suceso, descubre que se ha producido este intento y puede decidir qué acción es necesaria.
- Una aplicación intenta abrir un canal, pero el intento falla porque SSL no permite la conexión. Se emite un mensaje de suceso de instrumentación. Al inspeccionar el mensaje de suceso, descubre que se ha producido este intento y puede decidir qué acción es necesaria.

Conceptos relacionados

"Auditoría" en la página 7

La *auditoría* es el proceso de registrar y comprobar sucesos para detectar si ha tenido lugar una actividad no esperada o no autorizada, o si se ha llevado a cabo algún intento para realizar dicha actividad.

Confidencialidad en IBM MQ

Puede implementar la confidencialidad en IBM MQ cifrando mensajes.

La confidencialidad puede asegurarse en un entorno IBM MQ como se indica a continuación:

- Después de que un MCA emisor obtenga un mensaje de una cola de transmisión, IBM MQ utiliza TLS para cifrar el mensaje antes de enviarlo a través de la red al MCA receptor. En el otro extremo del canal, el mensaje se descifra antes de que el MCA receptor lo transfiera a la cola de destino.
- Mientras que los mensajes se almacenan en una cola local, los mecanismos de control de accesos proporcionados por IBM MQ se podrían considerar suficientes para proteger su contenido contra una revelación no autorizada. Sin embargo, para un mayor nivel de seguridad, puede utilizar Advanced Message Security para cifrar los mensajes almacenados en las colas.
- z/OS V 9.1.4 Los mensajes almacenados en las colas locales se pueden cifrar en reposo utilizando el cifrado de conjunto de datos de z/OS.

  Consulte la sección Confidencialidad para los datos en reposo en IBM MQ for z/OS con cifrado de conjunto de datos. para obtener más información.

Conceptos relacionados

"Confidencialidad" en la página 7

El servicio de *confidencialidad* protege la información confidencial para que no pueda divulgarse sin la autorización correspondiente.

Integridad de los datos en IBM MQ

Puede utilizar un servicio de integridad de datos para detectar si se ha modificado un mensaje.

La integridad de datos puede asegurarse en un entorno IBM MQ como se indica a continuación:

- Puede utilizar TLS para detectar si el contenido de un mensaje se ha modificado de forma deliberada mientras se transmitía a través de una red. En TLS, el algoritmo de resumen de mensaje proporciona la detección de mensajes modificados en tránsito.

  Todas las CipherSpecs de IBM MQ proporcionan un algoritmo de resumen de mensaje, excepto para TLS_RSA_WITH_NULL_NULL, que no proporciona integridad de los datos del mensaje.

  IBM MQ detecta los mensajes modificados al recibirlos; al recibir un mensaje modificado, IBM MQ emite un mensaje de error AMQ9661 y el canal se detiene.
- Mientras los mensajes se almacenan en una cola local, los mecanismos de control de accesos que proporciona IBM MQ pueden considerarse suficientes para impedir la modificación deliberada del contenido de los mensajes.

  Sin embargo, para un mayor nivel de seguridad, puede utilizar Advanced Message Security para detectar si el contenido de un mensaje se ha modificado deliberadamente entre la hora cuando se colocó el mensaje en la cola y la hora cuando se recuperó de la cola.

  Al detectar un mensaje modificado, la aplicación que intentaba recibir el mensaje recibe un código de retorno 2063 y, si se utiliza una llamada MQGET, el mensaje se mueve a SYSTEM.PROTECTION.ERROR.QUEUE

Conceptos relacionados

"Integridad de datos" en la página 7

El servicio de *integridad de datos* detecta si se han modificado los datos de forma no autorizada.

Cifrado en IBM MQ

IBM MQ proporciona cifrado utilizando el protocolo TLS (seguridad de la capa de transporte).

Para obtener más información, consulte “Protocolos de seguridad TLS en IBM MQ” en la página 24.

Conceptos relacionados

“Conceptos de cifrado” en la página 7
En esta colección de temas se describen los conceptos de cifrado aplicables a IBM MQ.

Protocolos de seguridad TLS en IBM MQ

IBM MQ da soporte al protocolo TLS (seguridad de la capa de transporte) para proporcionar seguridad a nivel de enlace para los canales de mensajes y los canales MQI.

Los canales de mensajes y los canales MQI pueden utilizar el protocolo TLS para proporcionar seguridad a nivel de enlace. Un MCA emisor es un cliente TLS y un MCA de respuesta es un servidor TLS. IBM MQ da soporte a TLS 1.0 y TLS 1.2. Puede especificar los algoritmos de cifrado que utiliza el protocolo TLS suministrando una CipherSpec como parte de la definición de canal.

Nota: A partir de IBM MQ 8.0.0 Fix Pack 2, se ha dejado de utilizar el protocolo SSLv3 y algunas CipherSpecs de IBM MQ. Para más información, consulte Deprecation: SSLv3 protocol.

Puede utilizar los parámetros SECPROT y SSLCIPH para mostrar el protocolo de seguridad y CipherSpec que se utilizan en un canal.

En cada extremo de un canal de mensajes y en el servidor de un canal MQI, el MCA actúa en nombre del gestor de colas al que está conectado. Durante el reconocimiento TLS, el MCA envía el certificado digital del gestor de colas a su MCA asociado en el otro extremo del canal. El código de IBM MQ en el extremo del cliente de un canal MQI actúa en nombre del usuario de la aplicación de cliente IBM MQ. Durante el reconocimiento TLS, el código IBM MQ envía el certificado digital del usuario al MCA en el extremo de servidor del canal MQI.

Los gestores de colas y los usuarios del cliente IBM MQ no necesitan tener certificados digitales personales asociados cuando actúan como clientes TLS, a menos que se especifique SSLCAUTH(REQUIRED) en el extremo del servidor del canal.

Los certificados digitales se almacenan en un *repositorio de claves*. El atributo de gestor de colas **SSLKeyRepository** especifica la ubicación del depósito de claves que contiene el certificado digital del gestor de colas. En un sistema de cliente IBM MQ, la variable de entorno MQSSLKEYR especifica la ubicación del repositorio de claves que contiene el certificado digital del usuario. De forma alternativa, la aplicación cliente IBM MQ puede especificar su ubicación en el campo **KeyRepository** de la estructura de opciones de configuración de TLS, MQSCO, en una llamada MQCONNX. Consulte los temas relacionados para obtener más información sobre los depósitos de claves y cómo especificar su ubicación.

Soporte para TLS

IBM MQ proporciona soporte para TLS 1.0 y TLS 1.2 según la plataforma que esté utilizando. Para obtener más información acerca del protocolo TLS, consulte la información en los subtemas.

IBM i
El soporte de TLS está integrado en el sistema operativo IBM i.

Clientes de Java y de JMS
Estos clientes utilizan JVM para proporcionar el soporte para TLS.

Sistemas UNIX, Linux, and Windows
El soporte de TLS se instala con IBM MQ.

z/OS
El soporte de TLS está integrado en el sistema operativo z/OS. El soporte de TLS en z/OS se conoce como *TLS del sistema*.

Para obtener información acerca de los requisitos previos del soporte para TLS en IBM MQ, consulte Requisitos del sistema para IBM MQ.

Conceptos relacionados

"Protocolos de seguridad de cifrado: TLS" en la página 15
Los protocolos de cifrado proporcionan conexiones seguras, permitiendo que dos partes se comuniquen con privacidad e integridad de datos. El protocolo TLS (Transport Layer Security) ha evolucionado a partir del protocolo SSL (Secure Sockets Layer). IBM MQ ofrece soporte para TLS.

Repositorio de claves SSL/TLS

Una conexión TLS mutuamente autenticada requiere un repositorio de claves en cada extremo de la conexión. El repositorio de claves incluye certificados digitales y claves privadas.

En esta información se utiliza el término general *depósito de claves* para describir el almacén de certificados digitales y sus claves privadas asociadas. Se hace referencia al repositorio de claves con diferentes nombres en diferentes plataformas y entornos que dan soporte a TLS:

- IBM i En IBM i: *almacén de certificados*
- En Java y JMS: *almacén de claves* y *almacén de confianza*
- ULW En UNIX, Linux, and Windows: *archivo de base de datos de claves*
- z/OS En z/OS: *conjunto de claves*

Para obtener más información, consulte "Certificados digitales" en la página 10 y "Conceptos de TLS (Transport Layer Security)" en la página 15.

Una conexión TLS mutuamente autenticada requiere un repositorio de claves en cada extremo de la conexión. El depósito de claves puede contener los siguientes certificados y solicitudes:

- Un número de certificados de CA de diversas autorizaciones de certificación que permiten al gestor de colas o cliente verificar los certificados que recibe de su asociado en el extremo remoto de la conexión. Los certificados individuales pueden estar dentro de una cadena de certificados.
- Uno o más certificados personales recibidos de una entidad emisora de certificados. Debe asociar un certificado personal diferente a cada gestor de colas o IBM MQ MQI client. Los certificados personales son esenciales en un cliente TLS si la autenticación mutua es necesaria. Si no se requiere autenticación mutua, los certificados personales no son necesarios en el cliente. El depósito de claves podría también contener la clave privada correspondiente a cada certificado personal.
- Las solicitudes de certificados que están en espera de ser firmados por un certificado de CA de confianza.

Para obtener más información acerca de cómo proteger su depósito de claves, consulte "Protección de repositorios de claves de IBM MQ" en la página 26.

La ubicación del repositorio de claves depende de la plataforma que esté utilizando:

IBM i **IBM i**

El depósito de claves es un almacén de certificados. El almacén de certificados del sistema predeterminado se encuentra en `/QIBM/UserData/ICSS/Cert/Server/Default` en el sistema de archivos integrado (IFS). IBM MQ almacena la contraseña para el almacén de certificados en un *archivo de ocultación de contraseña*. Por ejemplo, el archivo de ocultación para el gestor de colas `QM1` es `/QIBM/UserData/mqm/qmgrs/QM1/ssl/Stash.sth`.

De forma alternativa, puede especificar que el almacén de certificados del sistema IBM i se utilice en su lugar. Para ello, cambie el valor del atributo **SSLKEYR** del gestor de colas a `*SYSTEM`. Este valor indica que el gestor de colas debe utilizar el almacén de certificados del sistema, y el gestor de colas se registra para su uso como aplicación con el Gestor de certificados digitales (DCM).

El almacén de certificados también contiene la clave privada para el gestor de colas.

Sistemas UNIX, Linux, and Windows

El depósito de claves es un almacén de base de datos de claves. El nombre del archivo de base de datos de claves debe tener la extensión `.kdb`. Por ejemplo, en UNIX and Linux, el archivo de base de datos de claves predeterminado para el gestor de colas `QM1` es `/var/mqm/qmgrs/QM1/ssl/key.kdb`. Si IBM MQ está instalado en la ubicación predeterminada, la vía de acceso equivalente en Windows es `C:\ProgramData\IBM\MQ\Qmgrs\QM1\ssl\key.kdb`.

Cada archivo de base de datos de claves tiene un archivo de ocultación de contraseña asociado. Este archivo contiene las contraseñas codificadas que permiten a los programas acceder a la base de datos de claves. El archivo de ocultación de contraseña debe estar en el mismo directorio y tener la misma raíz de archivo que la base de datos de claves y debe acabar con el sufijo `.sth`, por ejemplo `/var/mqm/qmgrs/QM1/ssl/key.sth`.

Nota: Las tarjetas de hardware criptográfico PKCS #11 pueden contener los certificados y las claves que, de lo contrario, se guardan en un archivo de bases de datos de claves. Cuando los certificados y las claves se guardan en tarjetas PKCS #11, IBM MQ continúa necesitando acceso al archivo de base de datos de claves y a un archivo de ocultación de contraseña.

En sistemas UNIX y Windows, la base de datos de claves también contiene la clave privada para el certificado personal asociado con el gestor de colas o el IBM MQ MQI client.

z/OS

Los certificados se guardan en un conjunto de claves en z/OS.

Otros gestores de seguridad externos (ESM) también utilizan conjuntos de claves para almacenar certificados.

Las claves privadas las gestiona RACF.

Protección de repositorios de claves de IBM MQ

El repositorio de claves para IBM MQ es un archivo. Asegúrese de que solamente el usuario designado pueda acceder al archivo del repositorio de claves. Esto impedirá que un intruso o un usuario no autorizado pueda copiar el archivo del repositorio de claves en otro sistema y establezca, de este modo, un ID de usuario idéntico en dicho sistema para usurpar la identidad del usuario designado.

Los permisos de los archivos dependen del valor de umask del usuario y de qué herramienta se utiliza. En Windows, las cuentas de IBM MQ necesitan el permiso `BypassTraverseChecking` lo que significa que los permisos de las carpetas en la vía de acceso del archivo no tienen ningún efecto.

Compruebe los permisos de archivos de los archivos del repositorio de claves y asegúrese de que los archivos y la carpeta que los contiene no sean legibles por todos, preferiblemente ni siquiera legibles para grupos.

Hacer el almacén de datos de sólo lectura es una buena práctica, en cualquier sistema que utilice, dejando sólo al administrador como autorizado para habilitar operaciones de escritura para realizar el mantenimiento.

En la práctica, debe proteger todos los almacenes, independientemente de la ubicación y de si están protegidos por contraseña o no; proteja los repositorios de claves.

Etiquetas de certificados digitales, descripción de los requisitos

Al establecer TLS para utilizar certificados digitales, puede que tenga que cumplir algunos requisitos específicos para las etiquetas, en función de la plataforma utilizada y el método que utilice para la conexión.

¿Qué es la etiqueta de certificado?

Una etiqueta de certificado es un identificador exclusivo que representa un certificado digital almacenado en un depósito de claves y que proporciona un nombre legible adecuado con el que hace referencia a un certificado en concreto cuando se realizan funciones de gestión de claves. El usuario asigna la etiqueta de certificado cuando añade un certificado a un depósito de claves por primera vez.

La etiqueta de certificado está separada de los campos **Subject Distinguished Name** o **Subject Common Name** del certificado. Tenga en cuenta que **Subject Distinguished Name** y **Subject Common Name** son campos dentro del propio certificado. Se definen cuando se crea el certificado y no pueden cambiarse. No obstante, puede cambiar la etiqueta asociada a un certificado digital.

Sintaxis de la etiqueta de certificado

Una etiqueta de certificado puede contener letras, números y puntuación con las siguientes condiciones:

- Multi La etiqueta de certificado puede contener hasta 64 caracteres.
- z/OS La etiqueta de certificado puede contener hasta 32 caracteres.
- La etiqueta de certificado puede contener espacios.
- Las etiquetas son sensibles a las mayúsculas y minúsculas.
- En sistemas que utilizan EBCDIC katakana, no puede utilizar caracteres en minúsculas.

Los requisitos adicionales para los valores de etiqueta de certificado se especifican en las siguientes secciones.

¿Cómo se utiliza la etiqueta de certificado?

IBM MQ utiliza etiquetas de certificado para localizar un certificado personal que se envía durante el reconocimiento TLS. De esta manera se elimina la ambigüedad cuando hay más de un certificado personal en el depósito de claves.

Puede establecer la etiqueta de certificado en un valor de su elección. Si no establece un valor, se utiliza una etiqueta predeterminado que sigue un convenio de denominación en función de la plataforma que esté utilizando. Para obtener información detallada, consulte las secciones siguientes sobre plataformas concretas.

Notas:

1. No puede establecer la etiqueta de certificado por su cuenta en los sistemas Java o JMS.
2. Los canales autodefinidos creados mediante una salida de definición automática de canal (CHAD) no pueden establecer la etiqueta de certificado, ya que el reconocimiento TLS se produce en el momento en que se crea el canal. Establecer la etiqueta de certificado en una salida CHAD para los canales de entrada no tiene ningún efecto.

En este contexto, un cliente TLS hace referencia al asociado de la conexión que inicia el reconocimiento, que podría ser un cliente IBM MQ o bien otro gestor de colas.

Durante el reconocimiento TLS, el cliente TLS siempre obtiene y valida un certificado digital del servidor. Con la implementación de IBM MQ, el servidor TLS siempre solicita un certificado del cliente y éste siempre proporciona un certificado al servidor si encuentra uno. Si el cliente no puede localizar un certificado personal, el cliente envía una respuesta no `certificate` al servidor.

El servidor TLS siempre valida el certificado de cliente si se envía uno. Si el cliente no envía un certificado, la autenticación falla si el extremo del canal que actúa como servidor TLS se ha definido con el parámetro **SSLCAUTH** establecido en *REQUIRED* o un valor de parámetro **SSLPEER** establecido.

Tenga en cuenta que los canales de entrada (incluidos el receptor, el solicitante, el clúster receptor, el servidor no calificado y los canales de conexión con el servidor) sólo envían el certificado configurado si la versión de IBM MQ del igual remoto da soporte completo a la configuración de etiqueta de certificado y el canal utiliza una CipherSpecTLS.

Un canal de servidor no calificado es uno que no tiene establecido el campo CONNAME.

En todos los otros casos, el parámetro **CERTLABL** del gestor de colas determina el certificado enviado. En concreto, únicamente reciben el certificado configurado mediante el parámetro **CERTLABL** del gestor de colas los siguientes, independientemente del valor de etiqueta específico de canal:

- Antes de IBM MQ 9.1.1, todos los clientes Java y JMS actuales.

- V 9.1.1 A partir de clientes IBM MQ 9.1.1, Java y JMS que dan soporte a SNI (Server Name Indication), es decir, certificados canal por canal.
- Las versiones de IBM MQ anteriores a IBM MQ 8.0.
- Clientes .NET gestionados

Además, el certificado utilizado por un canal debe ser adecuado para la CipherSpec del canal; consulte “Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45 para obtener más información.

IBM MQ 8.0 y posteriores da soporte al uso de varios certificados en el mismo gestor de colas, utilizando una etiqueta de certificado por canal, especificada utilizando el atributo **CERTLABL** en la definición de canal. Los canales de entrada al gestor de colas (por ejemplo, conexión con servidor o receptor) se basan en detectar el nombre de canal utilizando la indicación de nombre de servidor (SNI) de TSL, a fin de presentar el certificado correcto del gestor de colas.

Si un canal se conecta al gestor de colas de destino a través de IBM MQ Internet Pass-Thru (MQIPT), y la ruta MQIPT tiene establecidos **SSLServer** y **SSLClient** , hay dos sesiones TLS separadas entre los puntos finales, y los datos SNI no fluyen a través de la interrupción de sesión. Esto impide que se utilice un certificado por canal en el gestor de colas de destino, para la conexión TLS entre MQIPT y el gestor de colas. Para utilizar un certificado por canal en el gestor de colas de destino, para una conexión TLS que pasa a través de MQIPT, la ruta MQIPT debe utilizar la modalidad de proxy TLS, que reenvía todos los flujos de control TLS intactos, incluido el nombre SNI. Para obtener más información sobre el soporte de TLS en MQIPT, consulte Soporte de SSL/TLS.

Los certificados que se utilizan para las conexiones TLS terminadas o iniciadas por MQIPT se pueden configurar individualmente para cada ruta, por ejemplo utilizando las propiedades de ruta **SSLServerSiteLabel** y **SSLClientSiteLabel**.

Si desea más información sobre cómo conectarse a un gestor de colas utilizando la autenticación unidireccional, es decir, cuando un cliente TLS no envía un certificado, consulte Conexión de dos gestores de colas utilizando la autenticación unidireccional.

Sistemas Multiplatforms

Multi

En Multiplatforms, el servidor TLS envía un certificado al cliente.

En el caso de gestores de colas y clientes respectivamente, se busca de forma secuencial un valor que no esté vacío en los siguientes orígenes: El primer valor que no esté vacío determina la etiqueta del certificado. La etiqueta del certificado debe existir en el repositorio de claves. Si no se encuentra un certificado coincidente cuyo formato y combinación de mayúsculas y minúsculas coincida con una etiqueta, se produce un error y el reconocimiento TLS fallará.

Gestores de colas

1. Atributo de etiqueta de certificado de canal **CERTLABL**.
2. Atributo de etiqueta de certificado de gestor de colas **CERTLABL**.
3. Un valor predeterminado con el formato: `ibmwebspheremq` al que se añade el nombre del gestor de colas, en minúsculas. Por ejemplo, para el gestor de colas `QM1`, la etiqueta de certificado predeterminada es `ibmwebspheremqqm1`.

Clientes de IBM MQ

1. Atributo de etiqueta de certificado **CERTLABL** en la definición de canal CLNTCONN.
2. Atributo de estructura MQSCO **CertificateLabel**.
3. Variable de entorno **MQCERTLABL**.
4. Atributo del archivo de cliente `.ini` (en su sección SSL) **CertificateLabel**

5. Un valor predeterminado con el formato: `ibmwebspheremq` al que se añade el ID de usuario de la aplicación de cliente que se está ejecutando, en minúsculas. Por ejemplo, para un ID de usuario USER1, la etiqueta de certificado predeterminada es `ibmwebspheremquser1`.

Sistemas z/OS

Los clientes de IBM MQ no están soportados en z/OS. Sin embargo, un gestor de colas de z/OS puede actuar con el rol de cliente TLS cuando inicia una conexión o de un servidor TLS cuando acepta una solicitud de conexión. Los requisitos de la etiqueta de certificado para gestores de colas de z/OS se aplican a ambos roles y son distintos de los requisitos en Multiplatforms.

En el caso de gestores de colas y clientes respectivamente, se busca de forma secuencial un valor que no esté vacío en los siguientes orígenes: El primer valor que no esté vacío determina la etiqueta del certificado. La etiqueta del certificado debe existir en el repositorio de claves. Si no se encuentra un certificado coincidente cuyo formato y combinación de mayúsculas y minúsculas coincida con una etiqueta, se produce un error y el reconocimiento TLS fallará.

1. Atributo de etiqueta de certificado de canal, **CERTLABL**.
2. Si se comparte, el atributo de etiqueta de certificado de grupo de compartición de cola, **CERTQSGL**.

   Si no se comparte, el atributo de etiqueta de certificado de gestor de colas **CERTLABL**.
3. Un valor predeterminado con el formato: `ibmWebSphereMQ` al que se añade el nombre del gestor de colas o del grupo de compartición de cola. Tenga en cuenta que esta serie distingue entre mayúsculas y minúsculas y debe escribir tal como se muestra. Por ejemplo, para el gestor de colas `QM1`, la etiqueta de certificado predeterminada es `ibmWebSphereMQQM1`.
4. Si no se encuentra un certificado con el formato en la opción “3” en la página 29, IBM MQ intenta utilizar el certificado marcado como predeterminado en el conjunto de claves.

Para obtener más información acerca de cómo visualizar el repositorio de claves, consulte “Localizar el repositorio de claves para un gestor de colas en z/OS” en la página 324.

Clientes de IBM MQ Java y de IBM MQ JMS

Los clientes de IBM MQ Java y de IBM MQ JMS utilizan los recursos de sus proveedores de JSSE (Java Secure Socket Extension) para seleccionar un certificado personal durante el reconocimiento TLS y, por lo tanto, no están sujetos a los requisitos de las etiquetas de certificados.

El comportamiento predeterminado es que el cliente JSSE examine los certificados del depósito de claves y seleccione el primer certificado personal aceptable que encuentre. Sin embargo, este comportamiento es sólo un valor predeterminado y depende de la implementación del proveedor de JSSE.

Además, la aplicación puede, en tiempo de ejecución, personalizar en gran medida la interfaz JSSE a través de la configuración y el acceso directo. Consulte la documentación que proporciona el proveedor JSSE para obtener detalles específicos.

Para la resolución de problemas o para comprender mejor el reconocimiento que realiza la aplicación de cliente IBM MQ Java en combinación con su proveedor JSSE específico, puede habilitar la depuración estableciendo `javax.net.debug=ssl` en el entorno de la JVM.

Puede establecer la variable en la aplicación durante la configuración o especificando `-Djavax.net.debug=ssl` en la línea de mandatos.

Renovación del depósito de claves del gestor de colas
Cuando se cambia el contenido de un repositorio de claves, el gestor de colas no recoge inmediatamente el contenido nuevo. Para que un gestor de colas utilice el nuevo contenido de repositorio de claves, debe emitir el mandato REFRESH SECURITY TYPE(SSL).

Este proceso tiene una intención, y evita que la situación en la que se ejecutan varios canales pueda utilizar distintas versiones de un repositorio de claves. Como control de seguridad, el gestor de colas sólo puede cargar una versión de repositorio de claves en cualquier momento.

Para obtener más información sobre el mandato REFRESH SECURITY TYPE(SSL), consulte REFRESH SECURITY.

También puede renovar un repositorio de claves utilizando mandatos PCF o IBM MQ Explorer. Para obtener más información, consulte el Mandato MQCMD_REFRESH_SECURITY y el tema *Renovación de la seguridad TLS* en la sección de IBM MQ Explorer de esta documentación de producto.

Conceptos relacionados

"Renovación de la vista de un cliente del contenido de repositorio de claves SSL/TLS y valores SSL/TLS" en la página 30
Para actualizar la aplicación cliente con el contenido renovado del repositorio de claves, debe detener y reiniciar la aplicación cliente.

Renovación de la vista de un cliente del contenido de repositorio de claves SSL/TLS y valores SSL/TLS
Para actualizar la aplicación cliente con el contenido renovado del repositorio de claves, debe detener y reiniciar la aplicación cliente.

No se puede renovar la seguridad en un cliente IBM MQ; no hay ningún equivalente al mandato REFRESH SECURITY TYPE(SSL) para clientes (consulte REFRESH SECURITY) para obtener más información.

Para actualizar la aplicación cliente con el contenido renovada del repositorio de claves, debe detener y reiniciar la aplicación, siempre que cambie el certificado de seguridad.

Si reiniciar el canal renueva la configuración, y si la aplicación tiene lógica de reconexión, es posible renovar la seguridad en el cliente emitiendo el mandato STOP CHL STATUS(INACTIVE).

Conceptos relacionados

"Renovación del depósito de claves del gestor de colas" en la página 29
Cuando se cambia el contenido de un repositorio de claves, el gestor de colas no recoge inmediatamente el contenido nuevo. Para que un gestor de colas utilice el nuevo contenido de repositorio de claves, debe emitir el mandato REFRESH SECURITY TYPE(SSL).

Protección por contraseña MQCSP

A partir de la IBM MQ 8.0, puede enviar contraseñas incluidas en la estructura MQCSP tanto si se han protegido mediante la función IBM MQ como si se han cifrado mediante el cifrado TLS.

Importante: La protección por contraseña MQCSP es útil para fines de prueba y desarrollo porque utilizar la protección por contraseña MQCSP es más sencillo que establecer el cifrado TLS, pero no es tan seguro. A efectos de producción, debe utilizar el cifrado TLS antes que la protección por contraseña IBM MQ, especialmente cuando la red entre el cliente y el gestor de colas no es confianza, ya que el cifrado TLS es más seguro.

Si le preocupa saber qué cifrado se está utilizando y cuánta protección ofrece, es necesario que utilice el cifrado TLS completo. En este caso, los algoritmos son públicamente conocidos y puede seleccionar el que sea adecuado para su empresa utilizando el atributo de canal **SSLCIPH**.

Para obtener más información sobre la estructura MQCSP, consulte Estructura MQCSP.

La protección por contraseña se utiliza cuando se cumplen las siguientes condiciones:

- Los dos extremos de la conexión utilizan IBM MQ 8.0 o posterior.
- El canal no está utilizando el cifrado TLS. Un canal no está utilizando el cifrado TLS si dicho canal tiene un atributo **SSLCIPH** en blanco o si el atributo **SSLCIPH** está establecido a una CipherSpec que no proporciona cifrado. Los cifrado nulos, por ejemplo, NULL_SHA, no proporcionan cifrado.
- Establece **MQCSP**.**AuthenticationType** a MQCSP_AUTH_USER_ID_AND_PWD. El establecimiento de este valor permite evaluar comprobaciones adicionales para decidir si se lleva a cabo la protección por contraseña. El valor predeterminado de **MQCSP**.**AuthenticationType** es MQCSP_AUTH_NONE. Con el valor predeterminado, no se lleva a cabo ninguna protección por contraseña. Para obtener más información, consulte **AuthenticationType**.
- Si el cliente es IBM MQ Explorer y la modalidad de compatibilidad de identificación de usuario no está habilitada, que no es el valor predeterminado. Esta condición solo es aplicable a IBM MQ Explorer.

Si estas condiciones no se cumplen, la contraseña se envía en texto sin formato a menos que lo prohíba el valor de configuración **PasswordProtection**.

El valor de configuración PasswordProtection

El atributo **PasswordProtection** en la sección Channels de los archivos de configuración .ini del cliente y del gestor de colas puede impedir que se envíen contraseñas como texto sin formato. El atributo puede adoptar uno de los valores siguientes. El valor predeterminado es `compatible`:

compatible
La contraseña puede enviarse como texto sin formato si el gestor de colas o el cliente está ejecutando una versión de IBM MQ anterior a la IBM MQ 8.0. Es decir, se permiten contraseñas de texto sin formato a efectos de compatibilidad.

Por lo tanto:

- La CipherSpec de TLS envía la contraseña cifrada si se utiliza el cifrado TLS y la CipherSpec no es nula.
- La contraseña se envía como texto sin formato si el gestor de colas o el cliente está ejecutando una versión de IBM MQ anterior a la IBM MQ 8.0 y el cifrado TLS no se utiliza. La contraseña se envía como texto sin formato porque versiones de IBM MQ anteriores a la IBM MQ 8.0 solo pueden enviar contraseñas como texto sin formato.
- La contraseña se envía protegida si tanto el gestor de colas como el cliente están ejecutando una versión de IBM MQ en IBM MQ 8.0 o posterior, y se utiliza una CipherSpec nula o no se utiliza el cifrado TLS. **MQCSP**.**AuthenticationType** debe establecerse en MQCSP_AUTH_USER_ID_AND_PWD.
- La conexión falla antes de que se envíe la contraseña si tanto el gestor de colas como el cliente están ejecutando una versión de IBM MQ en IBM MQ 8.0 o posterior, y **MQCSP**.**AuthenticationType** no está establecido en MQCSP_AUTH_USER_ID_AND_PWD.

always
La contraseña debe estar cifrada con una CipherSpec que no sea una CipherSpecnula o **MQCSP**.**AuthenticationType** debe establecerse en MQCSP_AUTH_USER_ID_AND_PWD. De lo contrario, la conexión fallará. Es decir, las contraseñas de texto sin formato no están permitidas.

Por lo tanto:

- La CipherSpec de TLS envía la contraseña cifrada si se utiliza el cifrado TLS y la CipherSpec no es nula.
- La contraseña se envía protegida si tanto el gestor de colas como el cliente ejecutan una versión de IBM MQ en IBM MQ 8.0 o posterior, y no se utiliza el cifrado TLS o se utiliza una CipherSpec nula. **MQCSP**.**AuthenticationType** debe establecerse en MQCSP_AUTH_USER_ID_AND_PWD.
- La conexión falla antes de que se envíe la contraseña si el gestor de colas o el cliente está ejecutando una versión de IBM MQ anterior a la IBM MQ 8.0 y el cifrado TLS no se utiliza. Como las versiones de IBM MQ anteriores a la IBM MQ 8.0 solo pueden enviar contraseñas como texto sin formato y `always` requiere que la contraseña esté cifrada o protegida, la conexión falla.

opcional
La contraseña se puede enviar opcionalmente protegida, pero se envía en texto sin formato si **MQCSP**.**AuthenticationType** no está establecido en MQCSP_AUTH_USER_ID_AND_PWD. Es decir, cualquier cliente puede enviar contraseñas de texto sin formato.

Por lo tanto:

- La CipherSpec de TLS envía la contraseña cifrada si se utiliza el cifrado TLS y la CipherSpec no es nula.
- La contraseña se envía en texto sin formato si se utiliza una CipherSpec nula y **MQCSP**.**AuthenticationType** no está establecido en MQCSP_AUTH_USER_ID_AND_PWD.
- La contraseña se envía como texto sin formato si el gestor de colas o el cliente está ejecutando una versión de IBM MQ anterior a la IBM MQ 8.0 y el cifrado TLS no se utiliza. La contraseña se envía

como texto sin formato porque versiones de IBM MQ anteriores a la IBM MQ 8.0 solo pueden enviar contraseñas como texto sin formato.

- La contraseña se envía protegida si tanto el gestor de colas como el cliente están ejecutando una versión de IBM MQ en IBM MQ 8.0 o posterior, no se utiliza el cifrado TLS o se utiliza una CipherSpec nula y **MQCSP**.**AuthenticationType** se establece en MQCSP_AUTH_USER_ID_AND_PWD.

warn
Cualquier cliente puede enviar contraseñas de texto sin formato. Si se recibe una contraseña de texto sin formato, se escribe un mensaje de aviso (AMQ9297) en los registros de error del gestor de colas.

Para los clientes Java y JMS, el comportamiento de los cambios de atributo **PasswordProtection** depende de la opción de utilizar la modalidad de compatibilidad o la modalidad MQCSP:

- Si los clientes Java y JMS están funcionando con la modalidad de compatibilidad, no fluye una estructura MQCSP durante el proceso de conexión. Por lo tanto, el comportamiento del atributo **PasswordProtection** es el mismo que el que se describe para los clientes ejecutan una versión de IBM MQ anterior a la IBM MQ 8.0.
- Si los clientes Java y JMS están funcionando con la modalidad MQCSP, el comportamiento del atributo **PasswordProtection** es el que se describe.

Para obtener más información sobre la autenticación de conexiones con clientes Java y JMS, consulte “Autenticación de conexión con el cliente Java” en la página 78.

gestor de certificados digitales (DCM)

Utilice DCM para gestionar certificados digitales y claves privadas en IBM i.

El Gestor de certificados digitales (DCM) le permite gestionar certificados digitales y utilizarlos en aplicaciones seguras en el servidor IBM i. Con el Gestor de certificados digitales, puede solicitar y procesar certificados digitales de Entidades emisoras de certificados (CA) o de terceros. También puede actuar como una entidad emisora de certificados local para crear y gestionar certificados digitales para sus usuarios.

DCM también da soporte al uso de Listas de revocación de certificados (CRL) para proporcionar un proceso de validación de certificados y aplicaciones más potente. Puede utilizar DCM para definir la ubicación donde reside una CRL de una entidad emisora de certificados específica en un servidor LDAP para que IBM MQ pueda verificar que no se ha revocado un certificado específico.

DCM da soporte y puede detectar automáticamente certificados en diversos formatos. Cuando DCM detecta un certificado codificado PKCS #12 o un certificado PKCS #7 que contiene datos cifrados, solicita automáticamente al usuario que escriba la contraseña que se ha utilizado para cifrar el certificado. DCM no solicita certificados PKCS #7 que no contengan datos cifrados.

DCM proporciona una interfaz de usuario basada en navegador que se puede utilizar para gestionar certificados digitales para las aplicaciones y los usuarios. La interfaz de usuario está dividida en dos secciones principales: una sección de navegación y una sección de tareas.

Utilice la sección de navegación para seleccionar las tareas para gestionar certificados o las aplicaciones que los utilizan. Algunas tareas individuales se muestran directamente en la sección de navegación principal, pero la mayoría de las tareas de la sección de navegación se organizan en categorías. Por ejemplo, Gestionar certificados es una categoría de tareas que contiene diversas tareas guiadas individuales, como por ejemplo Ver certificado, Renovar certificado e Importar certificado. Si un elemento de la sección de navegación es una categoría que contiene más de una tarea, se muestra una flecha a la izquierda del mismo. La flecha indica que cuando se selecciona el enlace de la categoría, se visualiza una lista ampliada de tareas, que le permite elegir qué tarea desea realizar.

Para obtener información importante sobre DCM, consulte las siguientes publicaciones IBM Redbooks:

- *IBM i Seguridad de red con conexión: OS/400 V5R1 Mejoras DCM y criptográficas*, SG24-6168. Concretamente, consulte los apéndices para obtener información esencial sobre la configuración del sistema IBM i como CA local.
- *AS/400 Internet Security: Developing a Digital Certificate Infrastructure*, SG24-5659. En concreto, véase el capítulo 5. *Digital Certificate Manager para AS/400* , que explica el AS/400 DCM.

Estándares federales de procesamiento de la información (FIPS)

En este tema se presenta los estándares federales de procesamiento de la información (FIPS) Cryptomodule Validation Program del US National Institute of Standards and Technology y las funciones de cifrado que se pueden utilizar en canales TLS.

Esta información se aplica a las siguientes plataformas:

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

ULW Para obtener más información sobre la conformidad con FIPS 140-2 de una conexión TLS de IBM MQ en UNIX, Linux, and Windows, consulte "Federal Information Processing Standards (FIPS) para UNIX, Linux, and Windows" en la página 33.

z/OS Para obtener más información sobre la conformidad con FIPS 140-2 de una conexión TLS de IBM MQ en z/OS, consulte "Federal Information Processing Standards (FIPS) para z/OS" en la página 36.

Si el hardware de cifrado está presente, los módulos de cifrado utilizados por IBM MQ se pueden configurar de modo que sean los proporcionados por el fabricante del hardware. En este caso, la configuración sólo será compatible con FIPS si dichos módulos de cifrado tienen certificación FIPS.

Con el tiempo, los Estándares federales de procesamiento de la información (FIPS) se actualizan para reflejar nuevos estándares frente a algoritmos y protocolos de cifrado. Por ejemplo, algunas CipherSpecs pueden dejar de certificarse con FIPS. Cuando se producen estos cambios, IBM MQ también se actualiza para implementar el último estándar. Como resultado, es posible que vea cambios en el comportamiento después de aplicar el mantenimiento.

Conceptos relacionados

"Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI" en la página 273
Cree sus propios repositorios de claves utilizando software compatible con FIPS y, a continuación, especifique que el canal debe utilizar las CipherSpecs certificadas por FIPS.

"Utilización de runmqckm, runmqakm y strmqikm para gestionar certificados digitales" en la página 290
En sistemas UNIX, Linux, and Windows, gestione las claves y los certificados digitales con la interfaz gráfica de usuario de **strmqikm** (iKeyman) o desde la línea de mandatos utilizando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

Tareas relacionadas

Habilitación de TLS en IBM MQ classes for Java

Utilización de TLS (seguridad de la capa de transporte) con IBM MQ classes for JMS

Referencia relacionada

Propiedades TLS de los objetos JMS

"Federal Information Processing Standards" en la página 20
El gobierno de EE.UU. ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. El NIST (National Institute for Standards and Technology) es un importante ente al que le preocupan las cuestiones relativas a los sistemas de TI y la seguridad. El NIST propone recomendaciones y genera estándares, entre los que se incluye FIPS (Federal Information Processing Standards).

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

Cuando se requiere criptografía en un canal SSL/TLS en sistemas Windows, UNIX and Linux, IBM MQ utiliza un paquete de criptografía denominado IBM Crypto for C (ICC). En las plataformas Windows, UNIX and Linux, , el software ICC ha pasado el programa de validación de módulo de cifrado de FIPS (estándares federales de procesamiento de la información) del Instituto Nacional de Estándares y Tecnología, en el nivel 140-2.

La conformidad con FIPS 140-2 de una conexión TLS de IBM MQ en sistemas Windows, UNIX and Linux es la siguiente:

- Para todos los canales de mensajes de IBM MQ message channels (excepto los tipos de canal CLNTCONN), la conexión es compatible con FIPS si se cumplen las condiciones siguientes:
  - Se ha certificado que la versión del GSKit ICC instalado cumple la norma FIPS 140-2 en la versión de sistema operativo y arquitectura de hardware instalada.
  - El atributo SSLFIPS del gestor de colas se ha establecido en YES.
  - Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.
- Para todas las aplicaciones IBM MQ MQI client, la conexión utiliza GSKit y es compatible con FIPS si se cumplen las condiciones siguientes:
  - Se ha certificado que la versión del GSKit ICC instalado cumple la norma FIPS 140-2 en la versión de sistema operativo y arquitectura de hardware instalada.
  - Ha especificado que sólo se utilizará el cifrado certificado por FIPS, tal como se describe en el tema relacionado para el cliente MQI.
  - Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.
- Para las aplicaciones de IBM MQ classes for Java que utilizan la modalidad de cliente, la conexión utiliza las implementaciones TLS de JRE y es compatible con FIPS si se cumplen las condiciones siguientes:
  - JRE (Java Runtime Environment) que se utiliza para ejecutar la aplicación cumple con la norma FIPS en la versión de sistema operativo y la arquitectura de hardware instalada.
  - Ha especificado que sólo se utilizará el cifrado certificado por FIPS, tal como se describe en el tema relacionado para el cliente Java.
  - Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.
- Para las aplicaciones de IBM MQ classes for JMS que utilizan la modalidad de cliente, la conexión utiliza las implementaciones TLS de JRE y es compatible con FIPS si se cumplen las condiciones siguientes:
  - JRE (Java Runtime Environment) que se utiliza para ejecutar la aplicación cumple con la norma FIPS en la versión de sistema operativo y la arquitectura de hardware instalada.
  - Ha especificado que sólo se utilizará el cifrado certificado por FIPS, tal como se describe en el tema relacionado para el cliente JMS.
  - Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.
- Para las aplicaciones cliente .NET no gestionadas, la conexión utiliza GSKit y es compatible con FIPS si se cumplen las condiciones siguientes:
  - Se ha certificado que la versión del GSKit ICC instalado cumple la norma FIPS 140-2 en la versión de sistema operativo y arquitectura de hardware instalada.
  - Ha especificado que sólo se utilizará el cifrado certificado por FIPS, tal como se describe en el tema relacionado para el cliente .NET.
  - Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.
- Para las aplicaciones cliente XMS .NET no gestionadas, la conexión utiliza GSKit y es compatible con FIPS si se cumplen las condiciones siguientes:
  - Se ha certificado que la versión del GSKit ICC instalado cumple la norma FIPS 140-2 en la versión de sistema operativo y arquitectura de hardware instalada.
  - Ha especificado que sólo se utilizará el cifrado certificado por FIPS, tal como se describe en la documentación XMS .NET.

– Todos los repositorios de claves se han creado y manipulado utilizando únicamente software compatible con FIPS como por ejemplo, **runmqakm** con la opción `-fips`.

Todas las plataformas soportadas cuentan con el certificado FIPS 140-2, excepto cuando se indique en el archivo Readme que se incluye con cada fixpack o paquete de actualización.

Para las conexiones TLS que utilizan GSKit, el componente que tiene el certificado de FIPS 140-2 se denomina *ICC*. Es la versión de este componente la que determina la conformidad FIPS de GSKit en cualquier plataforma determinada. Para determinar la versión de ICC instalada actualmente, ejecute el mandato **dspmqver -p 64 -v** .

A continuación se muestra un extracto de ejemplo de la salida de **dspmqver -p 64 -v** relacionada con 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.
@ (#) Reservados todos los derechos. 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:
```

La declaración de certificación NIST para GSKit ICC 8 (incluida en GSKit 8) se puede encontrar en la siguiente dirección: Programa de validación de módulo criptográfico.

Si el hardware de cifrado está presente, los módulos de cifrado utilizados por IBM MQ se pueden configurar de modo que sean los proporcionados por el fabricante del hardware. En este caso, la configuración sólo será compatible con FIPS si dichos módulos de cifrado tienen certificación FIPS.

Nota: Los clientes TLS y SSL de Solaris x86 de 32-bits configurados para la operación compatible con FIPS 140-2 fallan cuando se ejecutan en sistemas Intel. Este error se produce porque el archivo de biblioteca de GSKit-Crypto Solaris x86 de 32 bits compatible con FIPS 140-2 no se carga en el conjunto de chips de Intel. En los sistemas afectados, el error AMQ9655 se notifica en el registro de errores de cliente. Para resolver este problema, inhabilite la conformidad con FIPS 140-2 o recompile la aplicación cliente de 64-bits, porque el código de 64-bits no se ve afectado.

Restricciones de Triple DES aplicadas al operar en conformidad con FIPS 140-2

Cuando IBM MQ se ha configurado para que funcione en conformidad con FIPS 140-2, se aplican restricciones adicionales en relación con las CipherSpecs de Triple DES (3DES). Estas restricciones permiten la conformidad con la recomendación NIST SP800-67 de los Estados Unidos.

1. Todas las partes de la clave Triple DES deben ser exclusivas.
2. Ninguna parte de la clave Triple DES puede ser una clave débil, semi-débil o posiblemente débil de acuerdo con las definiciones de NIST SP800-67.
3. No pueden transmitirse más de 32 GB de datos por medio de la conexión antes de que se tenga que producir un restablecimiento de clave secreta. De forma predeterminada, IBM MQ no restablece la clave de sesión secreta, por lo que este restablecimiento se debe configurar. Si no habilita el restablecimiento de la clave secreta cuando se utiliza una CipherSpec Triple DES y la conformidad con FIPS 140-2 da como resultado el cierre de la conexión con el error AMQ9288 después de superar el número de bytes máximo. Para obtener información sobre cómo configurar el restablecimiento de la clave secreta, consulte “Restablecimiento de claves secretas SSL y TLS” en la página 452.

IBM MQ genera claves de sesión DES triple que ya cumplen con las reglas 1 y 2. Sin embargo, para satisfacer la tercera restricción, debe habilitar el restablecimiento de clave secreta cuando utilice Triple DES CipherSpecs en una configuración de FIPS 140-2. Como alternativa, puede evitar el uso de Triple DES.

Conceptos relacionados

"Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI" en la página 273
Cree sus propios repositorios de claves utilizando software compatible con FIPS y, a continuación, especifique que el canal debe utilizar las CipherSpecs certificadas por FIPS.

"Utilización de runmqckm, runmqakm y strmqikm para gestionar certificados digitales" en la página 290
En sistemas UNIX, Linux, and Windows, gestione las claves y los certificados digitales con la interfaz gráfica de usuario de **strmqikm** (iKeyman) o desde la línea de mandatos utilizando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

Tareas relacionadas

Habilitación de TLS en IBM MQ classes for Java

Utilización de TLS (seguridad de la capa de transporte) con IBM MQ classes for JMS

Referencia relacionada

Propiedades TLS de los objetos JMS

"Federal Information Processing Standards" en la página 20
El gobierno de EE.UU. ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. El NIST (National Institute for Standards and Technology) es un importante ente al que le preocupan las cuestiones relativas a los sistemas de TI y la seguridad. El NIST propone recomendaciones y genera estándares, entre los que se incluye FIPS (Federal Information Processing Standards).

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

Cuando se requiere criptografía en un canal SSL/TLS en z/OS, IBM MQ utiliza un servicio denominado System SSL. El objetivo de System SSL es proporcionar la capacidad de ejecutar de forma segura en una modalidad diseñada para adherirse al Federal Information Processing Standards (FIPS) Cryptomodule Validation Program del US National Institute of Standards and Technology, al nivel 140-2.

Al implementar conexiones compatibles con FIPS 140-2 con conexiones TLS de IBM MQ, debe tener en cuenta una serie de puntos:

- Para que los canales de mensajes de IBM MQ sean compatibles con FIPS, asegúrese de que se cumplen las condiciones siguientes:
  - El FMID de nivel 3 de seguridad de System SSL se ha instalado y configurado (consulte Planificación de la instalación de IBM MQ).
  - Los módulos de System SSL se han validado.
  - El atributo SSLFIPS del gestor de colas se ha establecido en **YES**.

Cuando se ejecuta en modalidad FIPS, System SSL utiliza CP Assist for Cryptographic Function (CPACF), cuando esté disponible. Las funciones de cifrado que lleva a cabo el hardware soportado por ICSF cuando se ejecuta en la modalidad no FIPS se siguen utilizando cuando se ejecuta en modalidad FIPS, con la excepción de la generación de firmas RSA, que debe realizarse en el software.

Tabla 2. Diferencias entre el soporte de algoritmos de la modalidad FIPS y la modalidad no FIPS.

	No FIPS		**FIPS**	
Algoritmo	**Tamaños de claves**	**Hardware**	**Tamaños de claves**	**Hardware**
RC2	40 y 128			
RC4	40 y 128			
DES	56	x		
TDES	168	x	168	x
AES	128 y 256	x	128 y 256	x

Tabla 2. Diferencias entre el soporte de algoritmos de la modalidad FIPS y la modalidad no FIPS. (continuación)

	No FIPS		FIPS	
Algoritmo	**Tamaños de claves**	**Hardware**	**Tamaños de claves**	**Hardware**
MD5	48			
SHA-1	160	x	160	x
SHA-2	224, 256, 384 y 512	x	224, 256, 384 y 512	x
RSA	512-4096	x	1024-4096	x
DSA	512-1024		1024	
DH	512-2048		2048	

En modalidad FIPS, System SSL sólo puede utilizar certificados que utilicen los algoritmos y tamaños de clave que se muestran en la Tabla 1. Durante la validación de certificados X.509 si se encuentra un algoritmo que es incompatible con la modalidad FIPS, el certificado no se puede utilizar y se trata como no válido.

Para las aplicaciones de clases IBM MQ que utilizan la modalidad de cliente en WebSphere Application Server , consulte Soporte de Federal Information Processing Standard.

Para obtener más información sobre la configuración del módulo System SSL, consulte System SSL Module Verification Setup.

Referencia relacionada

"Federal Information Processing Standards" en la página 20

El gobierno de EE.UU. ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. El NIST (National Institute for Standards and Technology) es un importante ente al que le preocupan las cuestiones relativas a los sistemas de TI y la seguridad. El NIST propone recomendaciones y genera estándares, entre los que se incluye FIPS (Federal Information Processing Standards).

Multi

Verificación de la configuración de TLS del gestor de colas con `mqcertck`

El mandato **MQCERTCK** es una herramienta para buscar errores comunes en la configuración de TLS del gestor de colas, y proporciona algunas sugerencias para resolver problemas.

Introducción

El mandato **`mqcertck`** comprueba lo siguiente:

- La existencia y los permisos del repositorio de claves del gestor de colas, al que se hace referencia en el atributo **SSLKEYR** del gestor de colas.
- La existencia y la validez del certificado del gestor de colas, al que se hace referencia en el atributo **CERTLABL** del gestor de colas.
- La existencia y la validez de los certificados a los que se hace referencia en los atributos **CERTLABL** del canal habilitado para TLS.
- El repositorio de claves y los certificados de las aplicaciones cliente, que incluye comprobar si los certificados están autorizados con el gestor de colas.

Nota: El mandato **`mqcertck`** no está disponible en z/OS o IBM i.

Utilización

Para utilizar el mandato **mqcertck**, ejecute el mandato `mqcertck`, junto con los parámetros necesarios, y los parámetros opcionales que necesite, desde una línea de mandatos.

Consulte mqcertck para obtener una descripción del mandato y de los parámetros que acepta el mandato.

Ejemplo

Ha terminado de configurar el gestor de colas `QM1` para permitir las conexiones TLS de los clientes que se conectan al canal SVRCONN del gestor de colas.

Utiliza la característica de múltiples certificados y, por lo tanto, tanto el gestor de colas como el canal tienen un etiqueta de certificado especificada en sus atributos **CERTLABL**. Al crear el canal, ha cometido un error en el atributo **CERTLABL** del canal, por lo que cuando un cliente intenta conectarse, el gestor de colas devuelve un código de retorno 2393 de MQRC_SSL_INITIALIZATION_ERROR.

Antes de activar el gestor de colas, debe utilizar el mandato **mqcertck** para verificar la configuración de TLS del gestor de colas.

Puede ejecutar el mandato `mqcertck QM1` y recibir la salida siguiente:

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

Esta salida le solicita que compruebe la definición de canal para el canal de conexión de servidor MQCERTCK.CHANNEL. Aquí verá el error que ha cometido y puede corregirlo antes de volver a ejecutar el mandato `mqcertck` para verificar que se ha resuelto el problema.

Verificación de las conexiones de cliente

El mandato **mqcertck** tiene la capacidad de verificar los repositorios de claves de cliente, así como la configuración TLS del gestor de colas. Para ello, **mqcertck** debe poder acceder al repositorio de claves del cliente desde la máquina que ejecuta el gestor de colas.

Al ejecutar el mandato **mqcertck** , si proporciona el parámetro **-clientkeyr** con la ubicación del repositorio de claves de cliente (excluyendo la extensión), **mqcertck** comprueba este repositorio de claves en el gestor de colas.

Si sabe qué canal utilizará el cliente para conectarse al gestor de colas, puede especificarlo con el distintivo **-clientchannel** .

Si el cliente está utilizando la autenticación mutua para conectarse al gestor de colas, puede utilizar el parámetro **-clientusername** o **-clientlabel** , para indicar al mandato **mqcertck** qué certificado debe utilizar en el repositorio de claves del cliente.

Si utiliza el certificado predeterminado y no proporciona una etiqueta de certificado a la aplicación cliente, puede utilizar los parámetros **-clientusername** y **username** que ejecutan esta aplicación.

Durante la operación del mandato **mqcertck** , el mandato genera la etiqueta de certificado `ibmwebspheremqXXXX` donde XXXX es el valor pasado en el parámetro **-clientusername** .

Para verificar completamente el repositorio de claves del cliente, el mandato **mqcertck** crea una conexión ficticia utilizando GSKit. Para ello, el mandato debe tener un puerto disponible al que pueda enlazarse durante las pruebas de cliente. El puerto predeterminado utilizado es 5857, pero si este ya está en uso puede especificar un puerto distinto que se utilizará durante las pruebas de cliente.

Nota: Aunque el mandato **mqcertck** se enlaza con un puerto, no se utiliza ninguna comunicación externa mediante **mqcertck** y todas las pruebas se realizan localmente.

SSL/TLS en el IBM MQ MQI client

IBM MQ ofrece soporte para TLS en los clientes. Puede adaptar el uso de TLS de varias maneras.

IBM MQ proporciona soporte TLS para IBM MQ MQI clients en sistemas Windows, UNIX and Linux. Si está utilizando IBM MQ classes for Java, consulte Utilización de IBM MQ classes for Java y si está utilizando IBM MQ classes for JMS, consulte Utilización de IBM MQ classes for JMS. El resto de esta sección no es aplicable a los entornos de Java o JMS.

Puede especificar el repositorio de claves para un IBM MQ MQI client con el valor MQSSLKEYR en el archivo de configuración de cliente IBM MQ, o cuando su aplicación realice una llamada MQCONNX. Dispone de tres opciones para especificar que un canal utiliza TLS:

- Utilizar una tabla de definiciones de canal
- Utilizar la estructura de opciones de configuración de SSL, MQSCO, en una llamada MQCONNX
- Utilizar Active Directory (en sistemas Windows)

No puede utilizar la variable de entorno MQSERVER para especificar que un canal utiliza TLS.

Puede seguir ejecutando las aplicaciones IBM MQ MQI client existentes sin TLS, siempre y cuando no se especifique TLS en el otro extremo del canal.

Si se efectúan cambios en una máquina cliente en el contenido del repositorio de claves TLS, la ubicación del repositorio de claves TLS, la información de autenticación o los parámetros de hardware de cifrado, debe finalizar todas las conexiones TLS para reflejar estos cambios en los canales de conexión de cliente que la aplicación utiliza para conectarse al gestor de colas. Una vez que hayan finalizado todas las conexiones, reinicie los canales TLS. Se utilizarán los nuevos valores TLS. Estos valores son análogos a los actualizados por el mandato REFRESH SECURITY TYPE(SSL) en los sistemas del gestor de colas.

Cuando el IBM MQ MQI client se ejecuta en un sistema Windows, UNIX and Linux con hardware de cifrado, debe configurar dicho hardware con la variable de entorno MQSSLCRYP. Esta variable es equivalente al parámetro SSLCRYP del mandato MQSC ALTER QMGR. Consulte ALTER QMGR para obtener una descripción del parámetro SSLCRYP del mandato ALTER QMGR MQSC. Si utiliza la versión GSK_PCS11 del parámetro SSLCRYP, la etiqueta de la señal PKCS #11 debe especificarse enteramente en minúsculas.

El restablecimiento de claves secretas TLS y FIPS reciben soporte en IBM MQ MQI clients. Para obtener más información, consulte los temas “Restablecimiento de claves secretas SSL y TLS” en la página 452 y“Federal Information Processing Standards (FIPS) para UNIX, Linux, and Windows” en la página 33.

Consulte “Configuración de la seguridad de IBM MQ MQI client” en la página 272 para obtener más información sobre el soporte TLS para los IBM MQ MQI clients.

Tareas relacionadas

Configuración de un cliente utilizando un archivo de configuración

Especificar que un canal MQI utiliza SSL/TLS
Para que un canal MQI utilice TLS, el valor del atributo *SSLCipherSpec*, del canal de conexión de cliente debe ser el nombre de un Ciphercliente IBM MQ en la plataforma de cliente.

Puede definir un canal de conexión de cliente con un valor para este atributo de las maneras siguientes. Se listan en el orden de prioridad descendente.

1. Cuando una salida Preconnect proporciona una estructura de definición de canal para utilizar.

   Una salida PreConnect puede proporcionar el nombre de un CipherSpec en el campo *SSLCipherSpec* de una estructura de definición de canal, MQCD. Esta estructura se devuelve en el campo **ppMQCDArrayPtr** de la estructura de parámetros de salida MQNXP utilizada por la salida PreConnect.
2. Cuando una aplicación cliente IBM MQ MQI client emite una llamada MQCONNX.

   La aplicación puede especificar el nombre de un CipherSpec en el campo *SSLCipherSpec* de una estructura de definición de canal, MQCD.Se hace referencia a esta estructura con la estructura de opciones de conexión MQCNO, que es un parámetro en la llamada MQCONNX.
3. Utilización de una tabla de definiciones de canal de cliente (CCDT).

   Una o varias entradas en una tabla de definiciones de canal de cliente pueden especificar el nombre de un CipherSpec.Por ejemplo, si crea una entrada mediante el mandato DEFINE CHANNEL MQSC, puede utilizar el parámetro SSLCIPH para especificar el nombre de un CipherSpec.
4. Utilización de Active Directory en Windows.

   En los sistemas Windows, puede utilizar el mandato de control **setmqscp** para publicar las definiciones de canal de conexión de cliente en Active Directory. Una o varias de estas definiciones pueden especificar el nombre de un CipherSpec.

Por ejemplo, si una aplicación cliente proporciona una definición de canal de conexión de cliente en una estructura MQCD en una llamada MQCONNX, esta definición se utilizará con preferencia a cualquier entrada de una tabla de definiciones de canal de cliente a la que el cliente IBM MQ puede acceder.

No puede utilizar la variable de entorno MQSERVER para proporcionar la definición de canal en el extremo cliente de un canal MQI que utiliza TLS.

Para comprobar si un certificado de cliente se ha transmitido, visualice el estado del canal en el extremo del servidor de un canal para saber si existe un valor de parámetro de nombre de igual.

Conceptos relacionados
"Especificación de una CipherSpec para un IBM MQ MQI client" en la página 441
Dispone de tres opciones para especificar una CipherSpec para un IBM MQ MQI client.

CipherSpecs y CipherSuites en IBM MQ
IBM MQ admite las CipherSpecs TLS 1.2 y los algoritmos RSA y Diffie-Hellman. Sin embargo, puede habilitar las CipherSpecs en desuso, si tiene que hacerlo.

Consulte "Habilitación de CipherSpecs" en la página 426 si desea información sobre:

- CipherSpecs admitidas por IBM MQ.
- Cómo se habilitan las CipherSpecs SSL 3.0 y TLS 1.0. en desuso

IBM MQ da soporte a los algoritmos de intercambio y autenticación de claves RSA y Diffie-Hellman. El tamaño de la clave que se utiliza durante el reconocimiento TLS puede depender del certificado digital que se utiliza, pero algunas CipherSpecs incluyen una especificación del tamaño de clave de reconocimiento. Los tamaños de clave de reconocimiento más grandes proporcionan una autenticación más fuerte. Con los tamaños de clave más pequeños, el reconocimiento es más rápido.

Conceptos relacionados
"CipherSpecs y CipherSuites" en la página 19
Los protocolos de seguridad criptográficos deben estar de acuerdo con los algoritmos utilizados por una conexión segura. CipherSpecs y CipherSuites definen combinaciones específicas de algoritmos.

NSA Suite B Cryptography en IBM MQ

En este tema se proporciona información sobre cómo configurar IBM MQ en Windows, Linux y UNIX para que se ajuste al perfil TLS 1.2 compatible con Suite B.

Con el tiempo, el estándar NSA Cryptography Suite B Standard se ha ido actualizando para reflejar nuevos ataques contra los algoritmos y protocolos de cifrado. Por ejemplo, algunos CipherSpecs pueden dejar de certificarse con Suite B. Cuando se producen estos cambios, IBM MQ también se actualiza para implementar el último estándar. Como resultado, es posible que vea cambios en el comportamiento después de aplicar el mantenimiento. El archivo readme de IBM MQ muestra la versión de Suite B implementada por cada nivel de mantenimiento de producto. Si configura IBM MQ para implantar la conformidad con Suite B, consulte siempre el archivo readme cuando planifique el mantenimiento. Consulte IBM MQ, WebSphere MQ, y los archivos léame del producto MQSeries.

En sistemas Windows, UNIX yLinux, IBM MQ se puede haber configurado para se ajuste al perfil TLS 1.2 compatible con Suite B, en los niveles de seguridad que figuran en la Tabla 1.

Tabla 3. Niveles de seguridad de Suite B con CipherSpecs y algoritmos de firma digital permitidos

Nivel de seguridad	**CipherSpecs permitidas**	**Algoritmos de firma digital permitidos**
128 bits	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-256 ECDSA con SHA-384
192 bits	ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-384
Ambos [1]	ECDHE_ECDSA_AES_128_GCM_SHA256 ECDHE_ECDSA_AES_256_GCM_SHA384	ECDSA con SHA-256 ECDSA con SHA-384

1. Es posible configurar los niveles seguridad de 128 bits y 192 bits simultáneamente. Dado que la configuración de Suite B determina los algoritmos de cifrado mínimos aceptables, la configuración de ambos niveles de seguridad es equivalente a configurar sólo el nivel de seguridad de 128 bits. Los algoritmos de cifrado del nivel de seguridad de 192 bits son más fuertes que el mínimo necesario para el nivel de seguridad de 128 bits, por lo que están permitidos para el nivel de seguridad de 128 bits incluso si el nivel de seguridad de 192 bits no está habilitado.

Nota: Los convenios de denominación que se utilizan para el Nivel de seguridad no representan necesariamente el tamaño de la curva o elíptica del tamaño de la clave del algoritmo de cifrado AES.

Compatibilidad de CipherSpec con Suite B

Aunque el comportamiento predeterminado de IBM MQ es no ajustarse al estándar de Suite B, IBM MQ puede configurarse para la conformidad con uno o ambos niveles de seguridad en sistemas Windows, UNIX and Linux. Tras la configuración satisfactoria de IBM MQ para utilizar la Suite B, cualquier intento de iniciar un canal de salida utilizando un CipherSpec que no cumpla el estándar Suite B generará el error AMQ9282. Esta actividad también hace que el cliente MQI devuelva el código de razón MQRC_CIPHER_SPEC_NOT_SUITE_B. De forma similar, si se intenta iniciar un canal de entrada utilizando una CipherSpec que no se ajusta a la configuración de Suite B, se produce el error AMQ9616.

Para obtener más información sobre CipherSpecs de IBM MQ, consulte “Habilitación de CipherSpecs” en la página 426.

Suite B y los certificados digitales

Suite B limita los algoritmos de firma digital que se pueden utilizar para firmar certificados digitales. Suite B también restringe el tipo de clave pública que puede contener certificados. Por lo tanto, se debe haber configurado IBM MQ para que utilice los certificados cuyo algoritmo de firma digital y tipo de clave pública que permita el nivel de seguridad de Suite B configurado del socio remoto. Se rechazan los certificados digitales que no cumplan los requisitos del nivel de seguridad y la conexión falla con el error AMQ9633 o AMQ9285.

Para el nivel de seguridad de Suite B de 128 bits, se necesita la clave pública del asunto de certificado para poder utilizar la curva elíptica NIST P-256 o NIST P-384 y que se haya firmado con la curva elíptica NIST P-256 o la curva o elíptica NIST P-384. En el nivel de seguridad de Suite B de 192 bits, se necesita la clave pública del asunto de certificado para poder utilizar la curva elíptica NIST P-384, y que se haya firmado con la curva elíptica NIST P-384.

Para obtener un certificado adecuado que funcione de forma compatible con Suite B, utilice el mandato **runmqakm** y especifique el parámetro **-sig_alg** para solicitar un algoritmo de firma digital adecuado. Los valores de parámetro EC_ecdsa_with_SHA256 y EC_ecdsa_with_SHA384 **-sig_alg** corresponden a las claves de curva elíptica firmadas por los algoritmos de firma digital de Suite B permitidos.

Para obtener más información sobre el mandato **runmqakm** , consulte opciones runmqckm y runmqakm.

Nota: Los mandatos **runmqckm** y **strmqikm** no dan soporte a la creación de certificados digitales para la operación compatible con Suite B.

Creación y solicitud de certificados digitales

Para crear un certificado digital autofirmado para probar Suite B, consulte “Creación de un certificado personal autofirmado en UNIX, Linux, and Windows” en la página 298.

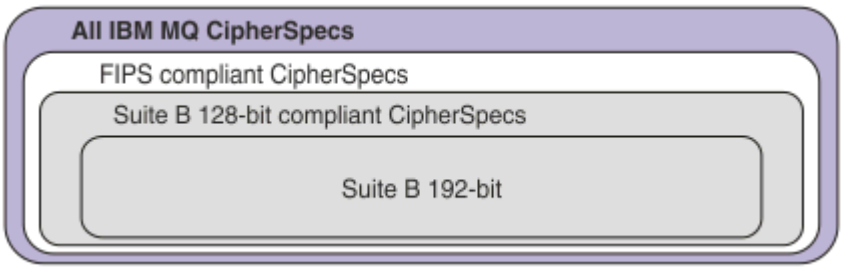
Para solicitar un certificado digital firmado por una CA para su utilización en la producción de Suite B, consulte “Solicitud de un certificado personal en UNIX, Linux, and Windows” en la página 301.

Nota: La entidad emisora de certificados que se utilice deberá generar certificados digitales que cumplan los requisitos descritos en IETF RFC 6460.

FIPS 140-2 y Suite B

El estándar Suite B es conceptualmente parecido a FIPS 140-2, ya que restringe el conjunto de algoritmos de cifrado permitidos para proporcionar un nivel de seguridad garantizado. Las CipherSpecs de Suite B soportadas actualmente se pueden utilizar cuando IBM MQ se ha configurado para que tenga un funcionamiento compatible con 140-2. Por consiguiente, es posible configurar IBM MQ para FIPS y Suite B de forma simultánea, en cuyo caso se aplican ambos conjuntos de restricciones.

El diagrama siguiente ilustra la relación entre estos subconjuntos:



Configuración de IBM MQ para el funcionamiento compatible con Suite B

Para obtener información sobre cómo configurar IBM MQ en Windows, UNIX and Linux para que funcione de forma compatible con Suite B, consulte “Configuración de IBM MQ para Suite B” en la página 43.

IBM MQ no da soporte al funcionamiento de forma compatible con Suite B en las plataformas IBM i and z/OS. Los clientes IBM MQ Java y JMS tampoco dan soporte al funcionamiento compatible con Suite B.

Conceptos relacionados

“Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI” en la página 273
Cree sus propios repositorios de claves utilizando software compatible con FIPS y, a continuación, especifique que el canal debe utilizar las CipherSpecs certificadas por FIPS.

Configuración de IBM MQ para Suite B

IBM MQ se puede configurar para que funcione en conformidad con el estándar Suite B de la NSA en las plataformas Windows, UNIX and Linux.

Suite B restringe el conjunto de algoritmos de cifrado permitidos para proporcionar un nivel de seguridad garantizado. IBM MQ se puede configurar para que funcione en conformidad con Suite B para proporcionar un nivel de seguridad mejorado. Para obtener más información sobre Suite B, consulte "Cifrado Suite B de la NSA (National Security Agency)" en la página 21. Para obtener más información sobre la configuración de Suite B y sus efectos sobre los canales TLS, consulte "NSA Suite B Cryptography en IBM MQ" en la página 41.

Gestor de colas

Para un gestor de colas, utilice el mandato **ALTER QMGR** con el parámetro **SUITEB** para establecer los valores adecuados para el nivel de seguridad que necesite. Para obtener más información, consulte ALTER QMGR.

También puede utilizar el mandato PCF **MQCMD_CHANGE_Q_MGR** con el parámetro **MQIA_SUITE_B_STRENGTH** para configurar el gestor de colas para que funcione de forma compatible con Suite B.

Nota: Si modifica valores de Suite B del gestor de colas, debe reiniciar el servicio MQXR para que estos valores entren en vigor.

Cliente MQI

De forma predeterminada, los clientes MQI no imponen la conformidad con Suite B. Puede habilitar la conformidad del cliente MQI para Suite B ejecutando una de las opciones siguientes:

1. Estableciendo el campo **EncryptionPolicySuiteB** en la estructura MQSCO en una llamada MQCONNX en uno o más de los valores siguientes:
   - MQ_SUITE_B_NONE
   - MQ_SUITE_B_128_BIT
   - MQ_SUITE_B_192_BIT

   No es válido utilizar MQ_SUITE_B_NONE con ningún otro valor.
2. Estableciendo la variable de entorno MQSUITEB en uno o varios de los valores siguientes:
   - NINGUNO
   - 128_BIT
   - 192_BIT

   Puede especificar varios valores utilizando una lista separada por comas. No es válido utilizar el valor NONE con ningún otro valor.
3. Estableciendo el atributo **EncryptionPolicySuiteB** de la stanza SSL del archivo de configuración de cliente MQI en uno o más de los valores siguientes:
   - NINGUNO
   - 128_BIT
   - 192_BIT

   Puede especificar varios valores utilizando una lista separada por comas. No es válido utilizar el valor NONE con ningún otro valor.

Nota: Los valores del cliente MQI se muestran en orden de prioridad. La estructura MSCO en la llamada MQCONNX altera temporalmente el valor en la variable de entorno MQSUITEB, que altera temporalmente el atributo en la stanza SSL.

Para obtener detalles completos sobre la estructura MQSCO, consulte MQSCO - Opciones de configuración de SSL.

Para obtener más información sobre la utilización de Suite B en el archivo de configuración de cliente, consulte Stanza SSL del archivo de configuración cliente.

Para obtener más información sobre el uso de la variable de entorno MQSUITEB, consulte Descripciones de variables de entorno.

.NET

Para los clientes no gestionados .NET, la propiedad **MQC.ENCRYPTION_POLICY_SUITE_B** indica el tipo de seguridad de Suite B necesaria.

Para obtener información sobre la utilización de Suite B en IBM MQ classes for .NET, consulte Clase MQEnvironment .NET.

AMQP

Los valores del atributo Suite B de un gestor de colas se aplican a los canales AMQP en dicho gestor de colas. Si modifica los valores de Suite B del gestor de colas, debe reiniciar el servicio AMQP para que los cambios entren en vigor.

Políticas de validación de certificados en IBM MQ

La política de validación de certificados determina controla el nivel de rigor con el que la validación de la cadena de certificados se ajusta a los estándares de la industria.

La política de validación de certificados depende de la plataforma y del entorno, tal como se indica a continuación:

- Para aplicaciones Java y JMS en todas las plataformas, la política de validación de certificados depende del componente JSSE del entorno de ejecución Java. Para obtener más información sobre la política de validación de certificados, consulte la documentación de su JRE.
- Para los sistemas IBM i, la política de validación de certificados depende de la biblioteca de sockets seguros proporcionada por el sistema operativo. Para obtener más información sobre la política de validación de certificados, consulte la documentación del sistema operativo.
- Para los sistemas z/OS, la política de validación de certificados depende del componente System SSL proporcionada por el sistema operativo. Para obtener más información sobre la política de validación de certificados, consulte la documentación del sistema operativo.
- Para los sistemas UNIX, Linux, and Windows, la política de validación de certificados la proporciona GSKit y puede configurarse. Hay dos políticas de validación de certificados diferentes admitidas:
  - Una política de validación de certificados existente, utilizada para la máxima compatibilidad e interoperatividad con certificados digitales anteriores que no cumplan con los estándares de validación de certificados IETF actuales. Esta política se conoce como política Básica.
  - Una política de validación de certificados estricta y compatible con los estándares que impone el estándar RFC 5280. Esta política se conoce como política Estándar.

Para obtener información sobre cómo configurar la política de validación de certificados en UNIX, Linux, and Windows, consulte “Configuración de políticas de validación de certificados en IBM MQ” en la página 44. Para obtener más información sobre las diferencias entre las políticas de validación de certificados básica y estándar, consulte la sección Validación de certificados y diseño de políticas de confianza en UNIX, Linux, and Windows.

Configuración de políticas de validación de certificados en IBM MQ

Puede especificar qué política de validación de certificado TLS se utiliza para validar certificados digitales recibidos de sistemas asociados remotos de cuatro modos.

En el gestor de colas, la política de validación de certificados se puede establecer de las siguientes formas:

- Utilizando el atributo del gestor de colas *CERTVPOL*. Si desea más información sobre cómo definir este atributo, consulte ALTER QMGR.

En el cliente, existen varios métodos que se pueden utilizar para establecer la política de validación de certificados. Si se utiliza más de un método para establecer la política, el cliente utiliza los valores en el siguiente orden de prioridad:

1. Utilizando el campo *CertificateValPolicy* en la estructura MQSCO del cliente. Si desea más información sobre cómo utilizar este campo, consulte MQSCO - opciones de configuración SSL.
2. Utilizando la variable de entorno, *MQCERTVPOL*. Si desea más información sobre cómo utilizar esta variable, consulte MQCERTVPOL.
3. Utilizando el parámetro de ajuste de la stanza SSL del cliente, *CertificateValPolicy*. Si desea más información sobre cómo utilizar este valor, consulte Stanza SSL del archivo de configuración de cliente.

Para obtener más información sobre las políticas de validación de certificados, consulte "Políticas de validación de certificados en IBM MQ" en la página 44.

Certificados digitales y compatibilidad de CipherSpec en IBM MQ

En este tema se proporciona información sobre cómo elegir las CipherSpecs y los certificados digitales adecuados para su política de seguridad, describiendo la relación entre las CipherSpecs y los certificados digitales en IBM MQ.

Únicamente un subconjunto de las CipherSpecs soportadas puede utilizarse con todos los tipos de certificados digitales soportados. Por consiguiente, es necesario que elija una CipherSpec adecuada para su certificado digital. Del mismo modo, si la política de seguridad de la organización requiere que utilice una CipherSpec determinada, debe obtener un certificado digital apropiado para dicha CipherSpec.

El algoritmo de firmas digitales MD5 y TLS 1.2

Los certificados digitales firmados mediante el algoritmo MD5 se rechazan cuando se utiliza el protocolo TLS 1.2. Esto se debe a que, ahora, muchos analistas consideran que el algoritmo MD5 es débil y, en general, se desaconseja su uso. Para utilizar CipherSpecs basadas en el protocolo TLS 1.2, asegúrese de que los certificados digitales no utilicen el algoritmo MD5 y sus firmas digitales. Las CipherSpecs que utilizan los protocolos TLS 1.0 no están sujetos a esta restricción y pueden continuar utilizando certificados con firmas digitales MD5.

Para ver el algoritmo de firma digital para un certificado determinado, puede utilizar el mandato **runmqakm** :

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

donde *etiqueta_certificado* es la etiqueta del certificado cuyo algoritmo de firma digital se ha de visualizar. Para obtener información detallada, consulte Etiquetas de certificados digitales.

Nota: Aunque la GUI de **runmqckm** (iKeycmd) y **strmqikm** (iKeyman) se pueden utilizar para ver una selección de algoritmos de firma digital, la herramienta **runmqakm** proporciona un rango más amplio.

La ejecución del mandato **runmqakm** genera una salida que muestra el uso del algoritmo de firma especificado:

```
Label : ibmmqexample
Key Size : 1024
Version : X509 V3
Serial : 4e4e93f1
Issuer : CN=Old Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : August 19, 2011 5:48:49 PM GMT+01:00
Not After : August 18, 2012 5:48:49 PM GMT+01:00
Public Key
    30 81 9F 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01
    05 00 03 81 8D 00 30 81 89 02 81 81 00 98 5A 7A
    F0 18 21 EE E4 8A 6E DE C8 01 4B 3A 1E 41 90 3D
    CE 01 3F E6 32 30 6C 23 59 F0 FE 78 6D C2 80 EF
    BC 83 54 7A EB 60 80 62 6B F1 52 FE 51 9D C1 61
    80 A5 1C D4 F0 76 C7 15 6D 1F 0D 4D 31 3E DC C6
    A9 20 84 6E 14 A1 46 7D 4C F5 79 4D 37 54 0A 3B
    A9 74 ED E7 8B 0F 80 31 63 1A 0B 20 A5 99 EE 0A
```

```
    30 A6 B6 8F 03 97 F6 99 DB 6A 58 89 7F 27 34 DE
    55 08 29 D8 A9 6B 46 E6 02 17 C3 13 D3 02 03 01
    00 01
Public Key Type : RSA (1.2.840.113549.1.1.1)
Fingerprint : SHA1 :
    09 4E 4F F2 1B CB C1 F4 4F 15 C9 2A F7 32 0A 82
    DA 45 92 9F
Fingerprint : MD5 :
    44 54 81 7C 58 68 08 3A 5D 75 96 40 D5 8C 7A CB
Fingerprint : SHA256 :
    3B 47 C6 E7 7B B0 FF 85 34 E7 48 BE 11 F2 D4 35
    B7 9A 79 53 2B 07 F5 E7 65 E8 F7 84 E0 2E 82 55
Signature Algorithm : MD5WithRSASignature (1.2.840.113549.1.1.4)
Value
    3B B9 56 E6 F2 77 94 69 5B 3F 17 EA 7B 19 D0 A2
    D7 10 38 F1 88 A4 44 1B 92 35 6F 3B ED 99 9B 3A
    A5 A4 FC 72 25 5A A9 E3 B1 96 88 FC 1E 9F 9B F1
    C5 E8 8E CF C4 8F 48 7B 0E A6 BB 13 AE 2B BD D8
    63 2C 03 38 EF DC 01 E1 1F 7A 6F FB 2F 65 74 D0
    FD 99 94 BA B2 3A D5 B4 89 6C C1 2B 43 6D E2 39
    66 6A 65 CB C3 C4 E2 CC F5 49 39 A3 8B 93 5A DD
    B0 21 0B A8 B2 59 5B 24 59 50 44 89 DC 78 19 51
Trust Status : Enabled
```

La línea `Signature Algorithm` muestra que se utiliza el algoritmo `MD5WithRSASignature` . Este algoritmo se basa en MD5 y por lo tanto este certificado digital no se puede utilizar con las CipherSpecs de TLS 1.2.

Interoperatividad de Elliptic Curve y CipherSpecs RSA

V 9.1.4 No se puede utilizar todas las CipherSpecs con todos los certificados digitales. Las CipherSpecs se indican mediante el prefijo de nombre CipherSpec . Cada tipo de CipherSpec impone diferentes restricciones sobre el tipo de certificado digital que se puede utilizar. Estas restricciones se aplican a todas las conexiones TLS de IBM MQ, pero resultan especialmente relevantes para los usuarios del cifrado Elliptic Curve.

La siguiente tabla resume las relaciones entre las CipherSpecs y los certificados digitales:

Tabla 4. Relaciones entre las CipherSpecs y los certificados digitales

Tipo	Prefijo de nombre de CipherSpec	Descripción	Tipo de clave pública necesaria	Algoritmo de cifrado de firma digital	Método de establecimiento de claves secretas
1	ECDHE_ECDSA_	CipherSpecs que utilizan claves públicas Elliptic Curve, claves secretas Elliptic Curve y algoritmos de firma digital Elliptic Curve.	Elliptic Curve	ECDSA	ECDHE
2	ECDHE_RSA_	CipherSpecs que utilizan claves públicas RSA, claves secretas de 4Elliptic Curve y algoritmos de firma digital RSA.	RSA	RSA	ECDHE
V 9.1.4 3	(All TLS 1.3 CipherSpecs)	CipherSpecs que utilizan claves públicas Elliptic Curve o RSA, claves secretas Elliptic Curve y algoritmos de firma digital Elliptic Curve o RSA.	Curva elíptica o RSA	ECDSA o RSA	ECDHE o RSA
4	(Todos los demás)	CipherSpecs que utilizan claves públicas RSA y algoritmos de firma digital RSA.	RSA	RSA	RSA

Nota: Los gestores de colas IBM MQ y los clientes MQI sólo dan soporte a las CipherSpecs de Tipo 1 y 2 en las plataformas IBM i.

En la columna de tipo de clave pública necesaria se muestra el tipo de clave pública que el certificado personal debe tener cuando utiliza cada tipo de CipherSpec. El certificado personal es el certificado de entidad final que identifica al gestor de colas o al cliente ante su socio remoto.

Puede configurar un canal con una CipherSpec que requiera un certificado EC (Elliptic Curve) y con una etiqueta de certificado para un certificado RSA o al revés. Debe asegurarse de que el certificado mencionado en la etiqueta de certificado sea adecuado para la CipherSpec del canal.

Presuponiendo que ha configurado correctamente IBM MQ, puede tener:

- Un gestor de colas individual con una combinación de certificados RSA y EC.
- Diferentes canales en el mismo gestor de colas que utilizan un certificado RSA o un certificado EC.

El algoritmo de cifrado de firma digital hace referencia al algoritmo de cifrado que se utiliza para validar al igual. El algoritmo de cifrado se utiliza junto con un algoritmo de hash como, por ejemplo, MD5, SHA-1 o SHA-256 para calcular la firma digital. Existen distintos algoritmos de firma digital que se pueden utilizar, por ejemplo, RSA con MD5 o ECDSA con SHA-256. En la tabla, ECDSA hace referencia al conjunto de algoritmos de firma digital que utilizan ECDSA; RSA hace referencia al conjunto de algoritmos de firma digital que utilizan RSA. Se puede utilizar cualquier algoritmo de firma digital soportado en el conjunto, siempre que se base en el algoritmo de cifrado indicado.

Las CipherSpecs de Tipo 1 requieren que el certificado personal tenga una clave pública Elliptic Curve. Cuando se utilizan estas CipherSpecs, se utiliza el acuerdo de claves Elliptic Curve Diffie Hellman Ephemeral para establecer la clave secreta de la conexión.

Las CipherSpecs de Tipo 2 requieren que el certificado personal tenga una clave pública RSA. Cuando se utilizan estas CipherSpecs, se utiliza el acuerdo de claves Elliptic Curve Diffie Hellman Ephemeral para establecer la clave secreta de la conexión.

Las CipherSpecs de tipo 3 requieren que el certificado personal tenga una clave pública RSA. Cuando se utilizan estas CipherSpecs, se utiliza el intercambio de claves RSA para establecer la clave secreta de la conexión.

Esta lista de restricciones no es exhaustiva: dependiendo de la configuración, puede haber restricciones adicionales que pueden afectar aún más a la capacidad de interoperar. Por ejemplo, si IBM MQ se ha configurado para cumplir los estándares FIPS 140-2 o Suite B de la NSA , esto también limitará el rango de configuraciones permitidas. Para obtener más información, consulte el siguiente apartado.

Si necesita utilizar diferentes tipos de CipherSpec en el mismo gestor de colas o aplicación de cliente, configure una combinación de etiqueta de certificado y CipherSpec en la definición de cliente.

Los tres tipos de CipherSpec no interactúan directamente: se trata de una limitación de los estándares actuales de TLS. Por ejemplo, supongamos que ha elegido utilizar ECDHE_ECDSA_AES_128_CBC_SHA256 CipherSpec para un canal receptor denominado TO.QM1 en un gestor de colas denominado QM1, el receptor debe tener un certificado personal con una clave Elliptic Curve y una firma digital basada en ECDSA. Si el canal receptor no cumple estos requisitos, el canal no se inicia.

Otros canales que se conecten al gestor de colas QM1 pueden utilizar otras CipherSpecs, siempre que cada canal utilice un certificado del tipo correcto para la CipherSpec de dicho canal. Por ejemplo, presuponga que QM1 utiliza un canal emisor denominado TO.QM2 para enviar mensajes a otro gestor de colas denominado QM2. El canal TO.QM2 puede utilizar la CipherSpec de Tipo 3 TLS_RSA_WITH_AES_256_CBC_SHA256 siempre que ambos extremos del canal utilicen certificados que contengan claves públicas RSA. Se puede utilizar el atributo de canal de etiqueta para configurar un certificado diferente para cada canal.

Cuando planifique sus redes IBM MQ, considere detenidamente qué canales requieren TLS y asegúrese de que el tipo de los certificados utilizados para cada canal sea adecuado para su uso con la CipherSpec en dicho canal.

Para ver el algoritmo de firma digital y el tipo de clave pública de un certificado digital, puede utilizar el mandato **runmqakm**:

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

donde *etiqueta_certificado* es la etiqueta del certificado cuyo algoritmo de firma digital necesita visualizar. Para obtener información detallada, consulte Etiquetas de certificados digitales.

La ejecución del mandato **runmqakm** generará una salida en la que se muestra el Tipo de clave pública:

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

En la línea Tipo de clave pública de este caso se muestra que el certificado tiene una clave pública Elliptic Curve. En la línea Algoritmo de firma de este caso se muestra que el algoritmo EC_ecdsa_with_SHA384 está en uso: se basa en el algoritmo ECDSA. Por tanto, este certificado sólo resulta adecuado para utilizarlo con las CipherSpecs de Tipo 1.

También puede utilizar el mandato **runmqckm** con los mismos parámetros. También se puede utilizar la GUI de **strmqikm** para ver algoritmos de firma digital si abre el repositorio de claves y efectúa una doble pulsación en la etiqueta del certificado. Sin embargo, debe utilizar la herramienta **runmqakm** para ver certificados digitales porque da soporte a un rango más amplio de algoritmos.

TLS 1.3 CipherSpecs

V 9.1.4

TLS 1.3 CipherSpecs da soporte a los certificados ECDSA y RSA.

CipherSpecs Elliptic Curve y Suite B de la NSA

Cuando se configura IBM MQ conforme al perfil TSL 1.2 compatible con Suite B, las CipherSpecs permitidas y los algoritmos de firma digital se restringen, tal como se describe en "NSA Suite B Cryptography en IBM MQ" en la página 41. Adicionalmente, el rango de claves Elliptic Curve aceptable se reduce, según los niveles de seguridad configurados.

En el nivel de seguridad de Suite B de 128 bits, se necesita la clave pública del asunto de certificado para poder utilizar la curva elíptica NIST P-256 o NIST P-384 y que se haya firmado con la curva elíptica NIST P-256 o la curva o elíptica NIST P-384. Se puede utilizar el mandato **runmqakm** para solicitar certificados digitales para este nivel de seguridad utilizando un parámetro -sig_alg de EC_ecdsa_with_SHA256 o EC_ecdsa_with_SHA384.

En el nivel de seguridad de Suite B de 192 bits, se necesita la clave pública del asunto de certificado para poder utilizar la curva elíptica NIST P-384, y que se haya firmado con la curva elíptica NIST P-384. Se puede utilizar el mandato **runmqakm** para solicitar certificados digitales para este nivel de seguridad utilizando un parámetro -sig_alg de EC_ecdsa_with_SHA384.

Las curvas elípticas NIST a las que se da soporte son las siguientes:

Tabla 5. Curvas elípticas NIST a las que se da soporte

Nombre de curva NIST FIPS 186-3	**Nombre de curva RFC 4492**	**Tamaño de clave de Elliptic Curve (bits)**
P-256	secp256r1	256
P-384	secp384r1	384
P-521	secp521r1	521

Nota: La curva elíptica NIST P-521 no se puede utilizar para el funcionamiento compatible con Suite B.

Conceptos relacionados

"Habilitación de CipherSpecs" en la página 426
Habilite una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC **DEFINE CHANNEL** o el mandato MQSC **ALTER CHANNEL**.

"Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI" en la página 273
Cree sus propios repositorios de claves utilizando software compatible con FIPS y, a continuación, especifique que el canal debe utilizar las CipherSpecs certificadas por FIPS.

"NSA Suite B Cryptography en IBM MQ" en la página 41
En este tema se proporciona información sobre cómo configurar IBM MQ en Windows, Linux y UNIX para que se ajuste al perfil TLS 1.2 compatible con Suite B.

"Cifrado Suite B de la NSA (National Security Agency)" en la página 21
El gobierno de los Estados Unidos de América ofrece asesoramiento técnico en cuanto a sistemas de TI y seguridad, que incluye el cifrado de datos. La NSA (National Security Agency) de Estados Unidos recomienda un conjunto de algoritmos de cifrado interoperables en su estándar Suite B.

Registros de autenticación de canal

Para ejercer un control más preciso sobre el acceso otorgado a la conexión de sistemas a nivel de canal, puede utilizar registros de autenticación de canal.

Un cliente puede intentar conectarse al gestor de colas utilizando un ID de usuario en blanco o un ID de usuario de alto nivel que permita al cliente realizar acciones malintencionadas. Puede bloquear el acceso a estos clientes utilizando registros de autenticación de canal. O bien, un cliente puede declarar un ID de usuario que sea válido en la plataforma del cliente, pero que sea desconocido o tenga un formato no válido en la plataforma del servidor. Puede utilizar un registro de autenticación de canal para correlacionar el ID de usuario declarado para un ID de usuario válido.

Puede encontrar una aplicación cliente que se conecta al gestor de colas y se comporta mal de alguna manera. Para proteger el servidor frente a los problemas que esta aplicación está causando, es necesario bloquearla temporalmente utilizando la dirección IP de la aplicación cliente hasta que se actualicen las reglas del cortafuegos o se corrija la aplicación cliente. Puede utilizar un registro de autenticación de canal para bloquear la dirección IP desde la que se conecta la aplicación cliente.

Si ha configurado una herramienta de administración tal como IBM MQ Explorer, y un canal para ese uso específico, puede asegurarse de que sólo puedan utilizarlo sistemas clientes determinados. Puede utilizar un registro de autenticación de canal para que el canal sólo pueda ser utilizado desde direcciones IP determinadas.

Si acaba de empezar con algunas aplicaciones de ejemplo que se ejecutan como cliente, consulte Preparación y ejecución de los programas de ejemplo para obtener un ejemplo de configuración del gestor de colas de forma segura utilizando registros de autenticación de canal.

Si desea obtener registros de autenticación de canal para controlar canales de entrada, utilice el mandato de MQSC **ALTER QMGR CHLAUTH(ENABLED)**.

Se aplican las reglas **CHLAUTH** para un MCA de canal que se crea en respuesta a una nueva conexión de entrada. Para un MCA de canal creado en respuesta al canal que se está iniciando localmente, no se aplica ninguna regla **CHLAUTH**.

Tabla 6. Dónde se aplican las reglas CHLAUTH para diferentes pares de canales

Tipo de canal	**MCA donde se aplican las reglas CHLAUTH**
SDR-RCVR	RCVR
RQSTR-SVR (iniciado en SVR)	RQSTR
RQSTR-SVR (iniciado en RQSTR)	SVR
RQSTR-SDR (iniciado en SDR)	RQSTR
RQSTR-SDR (iniciado en RQSTR)	SDR para la conexión inicial. RQSTR para la conexión de devolución de llamada.

Se pueden crear registros de autenticación de canal para realizar las funciones siguientes:

- Bloquear conexiones realizadas desde direcciones IP específicas.
- Bloquear conexiones realizadas desde identificadores de usuario específicos.
- Definir un valor MCAUSER para ser utilizado para cualquier canal que se conecte desde una dirección IP determinada.
- Definir un valor MCAUSER para ser utilizado para cualquier canal que declare un ID de usuario determinado.
- Definir un valor MCAUSER para ser utilizado para cualquier canal que tenga un determinado nombre distinguido de SSL o TLS.
- Definir un valor MCAUSER para ser utilizado para cualquier canal que se conecte desde un gestor de colas determinado.
- Bloquear conexiones que declaren proceder de un gestor de colas determinado, a menos que la conexión proceda de una dirección IP específica.
- Bloquear conexiones que presenten un certificado SSL o TLS determinado, a menos que la conexión proceda de una dirección IP específica.

Estos usos se explican con más detalle en las secciones siguientes.

Puede crear, modificar o eliminar registros de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH** o el mandato PCF **Set Channel Authentication Record**.

Nota: Un gran número de registros de autenticación de canal puede tener un impacto negativo en el rendimiento de un gestor de colas.

Bloqueo de direcciones IP

La función normal de un cortafuegos es impedir el acceso desde determinadas direcciones IP. No obstante, puede haber ocasiones en que se produzcan intentos de conexión desde una dirección IP que no debería tener acceso a su sistema IBM MQ y deberá bloquear la dirección temporalmente para que se pueda actualizar el cortafuegos. Es posible que estos intentos de conexión no procedan de canales

de IBM MQ; estos intentos de conexión puede que procedan de otras aplicaciones de socket que están mal configuradas para dirigirse a su escucha de IBM MQ. Puede bloquear direcciones IP estableciendo un registro de autenticación de canal de tipo BLOCKADDR. Puede especificar una o más direcciones individuales, rangos de direcciones, o patrones que incluyan caracteres comodín.

Cada vez que se rechaza una conexión de entrada porque la dirección IP se ha bloqueado de esta manera, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_ADDRESS, siempre que los sucesos de canal estén habilitados y el gestor de colas esté en ejecución. Además, la conexión se mantiene abierta durante 30 segundos antes de devolver el error para asegurar que el escucha no se vea inundado con repetidos intentos de conexión que están bloqueados.

Para bloquear direcciones IP sólo en canales específicos, o para evitar el retardo antes de notificar el error, establezca un registro de autenticación de canal de tipo ADDRESSMAP con el parámetro USERSRC(NOACCESS).

Cada vez que se rechaza una conexión de entrada por esta razón, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_NOACCESS, siempre que los sucesos de canal estén habilitados y el gestor de colas esté en ejecución.

Consulte “Bloquear direcciones IP específicas” en la página 390 para obtener un ejemplo.

Bloqueo de los ID de usuario

Para evitar que determinados identificadores de usuario se conecten a través de un canal de cliente, establezca un registro de autenticación de canal de tipo BLOCKUSER. Este tipo de registro de autenticación de canal se aplica sólo a los canales de cliente, no a los canales de mensajes. Puede especificar uno o más identificadores de usuario para bloquear, pero no puede utilizar comodines.

Cada vez que se rechaza una conexión de entrada por esta razón, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_USERID, siempre que los sucesos de canal estén habilitados.

Consulte “Bloquear identificadores (ID) de usuario específicos” en la página 392 para obtener un ejemplo.

Puede también bloquear cualquier acceso para identificadores de usuario especificados y determinados canales estableciendo un registro de autenticación de canal de tipo USERMAP mediante el parámetro USERSRC(NOACCESS).

Cada vez que se rechaza una conexión de entrada por esta razón, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_NOACCESS, siempre que los sucesos de canal estén habilitados y el gestor de colas esté en ejecución.

Consulte “Bloqueo del acceso de un ID de usuario cliente” en la página 395 para obtener un ejemplo.

Bloqueo de nombres de gestores de colas

Para bloquear el acceso a cualquier canal que se conecte desde un gestor de colas especificado, establezca un registro de autenticación de canal de tipo QMGRMAP con el parámetro USERSRC(NOACCESS). Puede especificar un nombre de gestor de colas individual o un patrón de caracteres que incluya comodines. No existe ningún homólogo de la función BLOCKUSER para bloquear el acceso para gestores de colas.

Cada vez que se rechaza una conexión de entrada por esta razón, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_NOACCESS, siempre que los sucesos de canal estén habilitados y el gestor de colas esté en ejecución.

Consulte “Bloquear el acceso desde un gestor de colas remoto” en la página 395 para obtener un ejemplo.

Bloqueo de nombres distinguidos de SSL o TLS

Para bloquear el acceso a cualquier usuario que declare un certificado personal SSL o TLS que contenga un nombre distinguido especificado, establezca un registro de autenticación de canal de tipo

SSLPEERMAP con el USERSRC(NOACCESS). Puede especificar un nombre distinguido individual o un patrón de caracteres que incluya comodines. No existe ningún homólogo de la función BLOCKUSER para bloquear el acceso para nombres distinguidos.

Cada vez que se rechaza una conexión de entrada por esta razón, se emite un mensaje de suceso MQRC_CHANNEL_BLOCKED con el calificador de razón MQRQ_CHANNEL_BLOCKED_NOACCESS, siempre que los sucesos de canal estén habilitados y el gestor de colas esté en ejecución.

Consulte "Bloquear el acceso para un Nombre distinguido SSL o TLS" en la página 396 para obtener un ejemplo.

Correlación de direcciones IP con los ID de usuario que se deben utilizar

Para especificar que cualquier canal que se conecte desde una dirección IP especificada debe utilizar un MCAUSER específico, establezca un registro de autenticación de canal de tipo ADDRESSMAP. Puede especificar una dirección individual, un rango de direcciones, o un patrón de caracteres que incluya comodines.

Si utiliza un reenviador de puertos, interruptor de sesión DMZ, o cualquier otra configuración que cambie la dirección IP presentada al gestor de colas, la correlación de direcciones IP puede no ser adecuada en su caso.

Consulte "Correlacionar una dirección IP con un ID de usuario MCAUSER" en la página 396 para obtener un ejemplo.

Correlación nombres de gestores colas con los ID de usuario que se deben utilizar

Para especificar que cualquier canal que se conecte desde un gestor de colas especificado debe utilizar un MCAUSER específico, establezca un registro de autenticación de canal de tipo QMGRMAP. Puede especificar un nombre de gestor de colas individual o un patrón de caracteres que incluya comodines.

Consulte "Correlacionar un gestor de colas remoto con un ID de usuario MCAUSER" en la página 393 para obtener un ejemplo.

Correlación de los ID de usuario declarados por un cliente con los ID de usuario que se deben utilizar

Para especificar que si una conexión de cliente de IBM MQ MQI utiliza un determinado ID de usuario debe utilizar un MCAUSER diferente especificado, establezca un registro de autenticación de canal de tipo USERMAP. La correlación de identificadores de usuario no utiliza comodines.

Consulte "Correlación de un ID de usuario cliente con un ID de usuario MCAUSER" en la página 394 para obtener un ejemplo.

Correlación de nombres distinguidos de SSL o TLS con los ID de usuario que se deben utilizar

Para especificar que cualquier usuario que declare un certificado personal SSL/TLS que contenga un nombre distinguido especificado debe utilizar un MCAUSER específico, establezca un registro de autenticación de canal de tipo SSLPEERMAP. Puede especificar un nombre distinguido individual o un patrón de caracteres que incluya comodines.

Consulte "Correlacionar un Nombre distinguido SSL o TLS con un ID de usuario MCAUSER" en la página 394 para obtener un ejemplo.

Correlación de gestores de colas, clientes o SSL o TLS DN de acuerdo con las direcciones IP

En algunos casos, un tercero puede suplantar un nombre de gestor de colas. También puede ser robado y reutilizado un certificado SSL o TLS o un archivo de base de datos de claves. Para protegerse contra estas amenazas, puede especificar que una conexión procedente de un gestor de colas o cliente determinado, o que utilice un nombre distinguido determinado se debe conectar desde una dirección IP especificada. Establezca un registro de la autenticación de canal de tipo USERMAP, QMGRMAP o SSLPEERMAP y

especifique la dirección IP permitida, o patrón de direcciones IP permitidas, utilizando el parámetro ADDRESS.

Consulte “Correlacionar un gestor de colas remoto con un ID de usuario MCAUSER” en la página 393 para obtener un ejemplo.

Interacción entre registros de autenticación de canal

Es posible que un canal que intenta establecer una conexión coincida con más de un registro de autenticación de canal, y que estos registros tengan efectos contradictorios. Por ejemplo, un canal puede declarar un ID de usuario que está bloqueado por un registro de autenticación canal BLOCKUSER, pero con un certificado SSL o TLS que coincida con un registro SSLPEERMAP que define un ID de usuario diferente. Además, si los registros de autenticación de canal utilizan comodines, una dirección IP, nombre de gestor de colas o nombre distinguido de SSL o TLS puede coincidir con varios patrones de caracteres. Por ejemplo, la dirección IP 192.0.2.6 coincide con los patrones 192.0.2.0-24, 192.0.2.*, y 192.0.*.6. La acción emprendida se determina de la forma siguiente.

- El registro de autenticación de canal utilizado se selecciona de la manera siguiente:
  - Un registro de autenticación de canal que coincida explícitamente con el nombre de canal tiene prioridad sobre un registro de autenticación de canal que coincida con el nombre de canal utilizando un comodín.
  - Un registro de autenticación de canal que haga uso de un nombre distinguido de SSL o TLS tiene prioridad sobre un registro que haga uso de un ID de usuario, nombre de gestor de colas o dirección IP.
  - Un registro de autenticación de canal que haga uso de un ID de usuario o nombre de gestor de colas tiene prioridad sobre un registro que haga uso de una dirección IP.
- Si se encuentra un registro de autenticación de canal coincidente y éste especifica un MCAUSER, este MCAUSER se asigna al canal.
- Si se encuentra un registro de autenticación de canal coincidente y éste especifica que el canal no tiene acceso, se asigna al canal un MCAUSER con el valor *NOACCESS. Este valor puede ser cambiado más tarde por un programa de salida de seguridad.
- Si no se encuentra ningún registro de autenticación de canal coincidente o se encuentra un registro coincidente y especifica que se debe utilizar el ID de usuario del canal, se examina el campo MCAUSER.
  - Si el campo MCAUSER está en blanco, se asigna el ID de usuario del cliente al canal.
  - Si el campo MCAUSER no está en blanco, su valor se asigna al canal.
- Se ejecuta cualquier programa de salida de seguridad. Este programa de salida puede establecer el ID de usuario del canal o determinar que se debe bloquear el acceso.
- Si se bloquea la conexión o MCAUSER está establecido en *NOACCESS, se cierra el canal.
- Si la conexión no se bloquea, para cualquier canal excepto un canal de cliente, se compara el ID de usuario de canal determinado en los pasos anteriores con la lista de usuarios bloqueados.
  - Si el ID de usuario está en la lista de usuarios bloqueados, el canal se cierra.
  - Si el ID de usuario no está en la lista de usuarios bloqueados, el canal se ejecuta.

Cuando varios registros de autenticación de canal coinciden con un nombre de canal, una dirección IP, un nombre de host, un nombre de gestor de colas o el nombre distinguido de SSL o TLS, se utiliza la coincidencia más específica. La coincidencia que se considera como:

- La más específica, es un nombre sin caracteres comodín, por ejemplo:
  - Un nombre de canal de A.B.C
  - Una dirección IP de 192.0.2.6
  - Un nombre de host de `hursley.ibm.com`
  - Un nombre de gestor de colas de 192.0.2.6
- La más genérica, es un solo asterisco (*) coincidente, por ejemplo:

  - Todos los nombres de canal
  - Todas las direcciones IP
  - Todos los nombres de host
  - Todos los nombres de gestores de colas
- Un patrón con un asterisco al principio de una serie es más genérico que un valor definido al principio de una serie:
  - Para los canales, *.B.C es más genérico que A.*
  - Para las direcciones IP, *.0.2.6 es más genérico que 192.*
  - Para los nombres de host, `*.ibm.com` es más genérico que `hursley.*`
  - Para los nombres de gestores de colas, *QUEUEMANAGER es más genérico que QUEUEMANAGER*
- Un patrón con un asterisco en un lugar específico en una serie es más genérico que un valor definido en el mismo lugar de una serie y, del mismo modo, para cada lugar posterior de una serie:
  - Para los canales, A.*.C es más genérico que A.B.*
  - Para las direcciones IP, 192.*.2.6 es más genérico que 192.0.*.
  - Para los nombres de host, `hursley.*.com` es más genérico que `hursley.ibm.*`
  - Para los nombres de gestores de colas, Q*MANAGER es más genérico que QUEUE*
- Cuando dos o más patrones tienen un asterisco en un lugar específico de una serie, el que tiene menos nodos después del asterisco es más genérico:
  - Para canales, A.* es más genérico que A.*.C
  - Para direcciones IP, 192.* es más genérico que 192.*.2.*.
  - Para los nombres de host, `hurlsey.*` es más genérico que `hursley.*.com`
  - Para los nombres de gestores de colas, Q* es más genérico que Q*MGR
- Adicionalmente, para una dirección IP:
  - Un rango indicado con un guión (-) es más específico que un asterisco. Por tanto, 192.0.2.0-24 es más específico que 192.0.2.*.
  - Un rango que es un subconjunto de otro mayor es más específico que el rango mayor. Por tanto, 192.0.2.5-15 es más específico que 192.0.2.0-24.
  - No están permitidos los rangos solapados. Por ejemplo, no puede tener registros de autenticación de canal para 192.0.2.0-15 y 192.0.2.10-20 al mismo tiempo.
  - Un patrón no puede tener menos números de componentes que los necesarios a no ser que el patrón termine con un asterisco individual final. Por ejemplo, 192.0.2 no es válido, pero 192.0.2.* es válido.
  - Un asterisco final debe separarse del resto de la dirección mediante el separador de parte adecuado (un punto (.) para IPv4, dos puntos (:) para IPv6). Por ejemplo, 192.0* no es válido porque el asterisco no está separado.
  - Un patrón puede contener asteriscos adicionales, a condición de que no haya ningún asterisco adyacente al asterisco final. Por ejemplo, 192.*.2.* es válido, pero 192.0.*.* no es válido.
  - Un patrón de dirección IPv6 no puede contener un signo doble de dos puntos y un asterisco final, porque la dirección resultante será ambigua. Por ejemplo, 2001::* podría expandirse a 2001:0000:*, 2001:0000:0000:* etc.
- Para un nombre distinguido de SSL o TLS, el orden de prioridad de las subseries de caracteres es el siguiente:

Tabla 7. Orden de prioridad de subseries

Orden	**Subserie de nombre distinguido**	**Nombre**
1	SERIALNUMBER=	Número de serie de certificado

Tabla 7. Orden de prioridad de subseries (continuación)

Orden	Subserie de nombre distinguido	Nombre
2	MAIL=	Dirección de correo electrónico
3	E=	Dirección de correo electrónico (En desuso por ser preferible MAIL)
4	UID=, USERID=	Identificador de usuario
5	CN=	Nombre común
6	T =	Título
7	OU=	Unidad organizativa
8	DC=	Componente de dominio
9	O=	Organización
10	STREET=	Calle / Primera línea de dirección
11	L=	Localidad
12	ST=, SP=, S=	Nombre del estado o provincia
13	P=	Código postal
14	C=	País
15	UNSTRUCTUREDNAME=	Nombre de host
16	UNSTRUCTUREDADDRESS=	Dirección IP
17	DNQ=	Calificador de nombre distinguido

Por tanto, si un certificado SSL o TLS se presenta con un DN que contenga las subseries O=IBM y C=UK, IBM MQ utiliza preferentemente un registro de autenticación de canal para O=IBM, en vez de uno para C=UK, si ambos están presentes.

Un nombre distinguido puede contener varias OU, que se deben especificar en orden jerárquico con las unidades organizativas más grandes especificadas primero. Si dos nombres distinguidos son iguales en todos los sentidos excepto por sus valores de unidad organizativa, el nombre distinguido más específico se determina de la siguiente manera:

1. Si tienen diferentes números de atributos de unidad organizativa, el nombre distinguido con más valores de unidad organizativa es más específico. La razón es que el nombre distinguido con más unidades organizativas califica más en detalle al nombre distinguido y proporciona más criterios de coincidencia. Aunque la unidad organizativa de nivel superior fuera un asterisco (OU=*), el nombre distinguido con más unidades organizativas sigue considerándose como el más específico.
2. Si tienen el mismo número de atributos de unidad organizativa, los pares correspondientes de valores de unidad organizativa se comparan en secuencia, de izquierda a derecha, donde la unidad organizativa más a la izquierda es el nivel superior (menos específica), de acuerdo con las reglas siguientes:
   a. Una unidad organizativa sin valores de asterisco es la más específica porque sólo puede coincidir con una serie.
   b. Una unidad organizativa con un único asterisco al principio o al final (por ejemplo, OU=ABC* o OU=*ABC) es la siguiente más específica.
   c. Una unidad organizativa con dos asteriscos, (por ejemplo OU=*ABC*) es la siguiente más específica.

d. Una unidad organizativa formada sólo por un asterisco (OU=*) es la menos específica.

3. Si la comparación de series es entre dos valores de atributo de la misma especificidad, la serie del atributo más largo es más específica.
4. Si la comparación de series es entre dos valores de atributo de la misma especificidad y longitud, se comparan las series (sin tener en cuenta mayúsculas y minúsculas) de la parte del nombre distinguido excluidos los asteriscos.

Si dos nombres distinguidos son iguales en todos los aspectos excepto en sus valores de DC, se aplican las mismas reglas de coincidencia que para las OU, excepto que en los valores de DC, el DC más izquierda es el nivel más bajo (más específico) y el orden de comparación difiere en consecuencia.

Visualización de registros de autenticación de canal

Para visualizar registros de autenticación de canal, utilice el mandato MQSC **DISPLAY CHLAUTH** o el mandato PCF **Inquire Channel Authentication Records**. Puede obtener todos los registros que coincidan con el nombre de canal proporcionado, o puede buscar una coincidencia explícita. La coincidencia explícita le indica qué registro de autenticación de canal se utilizará si un canal intenta establecer una conexión desde una dirección IP específica, desde un gestor de colas específico o utilizando un ID de usuario específico y, opcionalmente, que declare un certificado personal SSL/TLS que contenga un nombre distinguido especificado.

Conceptos relacionados

"Seguridad de la mensajería remota" en la página 96

En este apartado se tratan aspectos relativos a la seguridad de la mensajería remota.

Interacción de CHLAUTH y CONNAUTH

Cómo interactúan los registros de autenticación de canal (CHLAUTH) y la autenticación de conexión (CONNAUTH) en IBM MQ, en el caso de una única conversación en un canal.

Distintos tipos de enlaces

IBM MQ admite dos métodos para una aplicación para conectarse:

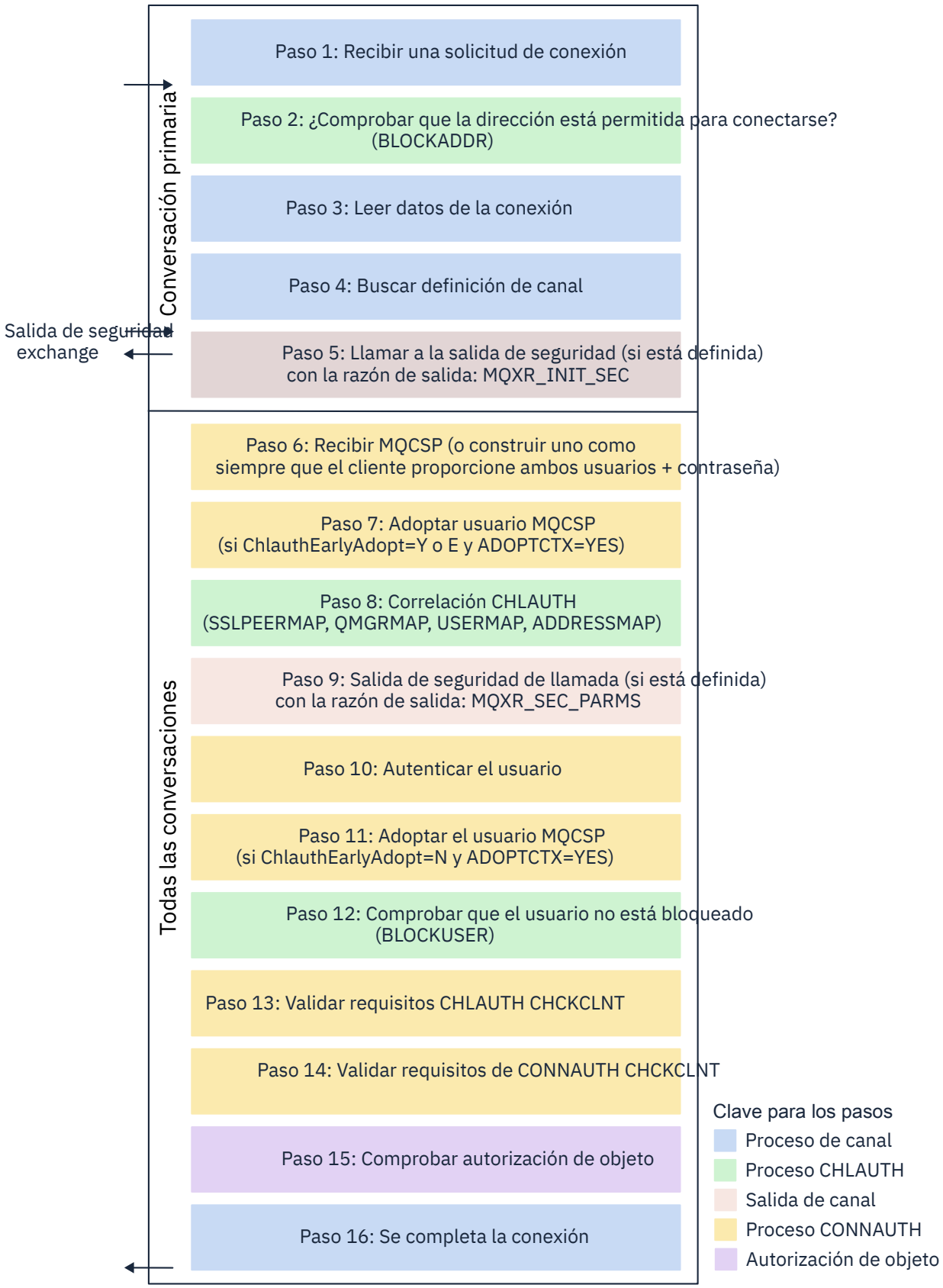
Enlaces locales
Se aplica cuando la aplicación y el gestor de colas están en la misma imagen operativa. CHLAUTH no es relevante para este tipo de conexión de aplicación.

Enlaces de cliente
Se aplica cuando la aplicación y el gestor de colas utilizan la red para comunicarse. La aplicación y el gestor de colas se pueden ejecutar en la misma máquina, o pueden estar en máquinas diferentes. En IBM MQ, una conexión de cliente se maneja en forma de un canal de conexión-servidor (SVRCONN) y, en esta situación, son aplicables ambos, CONNAUTH y CHLAUTH.

Pasos de enlaces del extremo de recepción de un canal

Cuando una aplicación se conecta a un gestor de colas, se realiza una cantidad considerable de comprobaciones para asegurarse de que ambos extremos del canal comprenden qué soporta el otro extremo. El extremo de recepción del canal realiza algunas comprobaciones adicionales, que implican CHLAUTH y CONNAUTH, para asegurarse de que el cliente está autorizado para conectarse, y este proceso también podría incluir una salida de seguridad, ya que esto puede afectar al resultado. También se ha referencia a esta fase de conexión del canal como la *fase de enlace*.

El diagrama siguiente lista los pasos que sigue un canal SVRCONN cuando se inicia el extremo del servidor (en el gestor de colas):



Paso 1: Recibir una solicitud de conexión
El iniciador de canal o el escucha recibe una solicitud de conexión de algún lugar de la red.

Paso 2: ¿La dirección está autorizada para conectarse?
Antes de que se pueda leer cualquier dato, IBM MQ comprueba la dirección IP del socio con respecto a las reglas CHLAUTH, para ver si la dirección está en la regla *BLOCKADDR*. Si la dirección no se encuentra y, por lo tanto, no está bloqueada, el flujo continúa hasta el siguiente paso.

Paso 3: Leer datos del canal
Ahora IBM MQ lee los datos en un almacenamiento intermedio y empieza a procesar la información enviada.

Paso 4: Buscar la definición de canal
En el primer flujo de datos, IBM MQ envía, entre otras cosas, el nombre del canal que está intentando iniciar el extremo del servidor. El gestor de colas de recepción busca la definición de canal, que tiene todos los valores que se han especificado para el canal.

Paso 5: Llamar a la salida de seguridad (si hay alguna definida)
Si el canal tiene definida una salida de seguridad (SCYEXIT), se llama con la razón de salida (MQCXP.**ExitReason**) establecer en MQXR_INIT_SEC.

Paso 6: Recibir MQCSP
Si es necesario, construya uno, siempre que el cliente proporcione el ID de usuario y la contraseña.

Si el cliente es una aplicación Java o JMS que se ejecuta en modalidad de compatibilidad, el cliente no pasa una estructura MQCSP al gestor de colas. En su lugar, si la aplicación ha suministrado un ID de usuario y una contraseña, se genera aquí una estructura MQCSP.

Paso 7: Adopte el usuario MQCSP (si `ChlauthEarlyAdopt` es *Y* y ADOPTCTX=YES)
Se autentica el ID de usuario certificado por el cliente.

Si CONNAUTH utiliza LDAP para correlacionar un nombre distinguido certificado con un ID de usuario corto, se produce la correlación en este paso.

Si la autenticación resulta satisfactoria, el canal adopta el ID de usuario y se utiliza en el paso de correlación CHLAUTH.

Nota: A partir de IBM MQ 9.0.4 se añade automáticamente el parámetro **ChlauthEarlyAdopt=** *Y* a la stanza de canales del archivo qm.ini para nuevos gestores de colas.

Paso 8: Correlación CHLAUTH
Se vuelve a inspeccionar la memoria caché CHLAUTH para buscar las reglas de correlación *SSLPEERMAP*, *USERMAP*, *QMGRMAP* y *ADDRESSMAP*.

Se utiliza la regla que coincide de forma más específica con el canal entrante. Si la regla tiene **USERSRC***(CHANNEL)* o *(MAP)*, el canal continúa en el enlace.

Si las reglas CHLAUTH se evalúan en una regla con **USERSRC***(NOACCESS)*, se bloquea la conexión de la aplicación al canal, a no ser que se sustituyan las credenciales posteriormente con un ID de usuario y una contraseña válidos en el paso 9.

Paso 9: Llamar a la salida de seguridad (si hay alguna definida)
Si el canal tiene definida una salida de seguridad (SCYEXIT), se llama con la razón de salida (MQCXP.**ExitReason**) establecer en MQXR_SEC_PARMS.

Estará presente un puntero a MQCSP en el campo **SecurityParms** de la estructura MQCXP.

La estructura MQCSP tiene punteros al ID de usuario (MQCSP.**CSPUserIdPtr**) y contraseña (MQCSP.**CSPPasswordPtr**).

Es posible cambiar el ID de usuario y la contraseña en la salida. En el ejemplo siguiente se muestra cómo imprimiría una salida de seguridad los valores de ID de usuario y contraseña en un registro de auditoría:

```
if (pMQCXP -> ExitReason == MQXR_SEC_PARMS)
{
  /* It is not a good idea for security reasons to print out the user ID */
  /* and password but the following is shown for demonstration reasons */
  printf("User ID: %.*s Password: %.*s\n",
         pMQCXP -> SecurityParms -> CSPUserIdLength,
```

```
        pMQCXP -> SecurityParms -> CSPUserIdPtr,
        pMQCXP -> SecurityParms -> CSPPasswordLength,
        pMQCXP -> SecurityParms -> CSPPasswordPtr);
```

La salida puede indicar a IBM MQ que cierre el canal, devolviendo *MQXCC_CLOSE_CHANNEL* en MQCXP.Campo **Exitresponse** . De lo contrario, el proceso del canal continúa hasta la fase de autenticación de conexión.

Nota: Si la salida de seguridad cambia el usuario certificado, no se volverán a aplicar las reglas de correlación CHLAUTH al nuevo usuario.

Paso 10: Autenticar el usuario

La fase de autenticación se produce si CONNAUTH está habilitado en el gestor de colas.

Para comprobar esto, emita el mandato MQSC 'DISPLAY QMGR CONNAUTH'.

z/OS El ejemplo siguiente muestra la salida del mandato **DISPLAY QMGR CONNAUTH** de un gestor de colas que se ejecuta en IBM MQ for z/OS.

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

Multi El ejemplo siguiente muestra la salida del mandato '**DISPLAY QMGR CONNAUTH**' de un gestor de colas que se ejecuta en IBM MQ for Multiplatforms.

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

El valor de CONNAUTH es el nombre de un objeto **AUTHINFO** IBM MQ.

Puesto que la autenticación del sistema operativo (**AUTHTYPE**(*IDPWOS*)) es válida en ambos sistemas, IBM MQ for Multiplatforms y IBM MQ for z/OS, el ejemplo utiliza la autenticación del sistema operativo.

z/OS El ejemplo siguiente muestra el objeto predeterminado proporcionado para **AUTHTYPE**(*IDPWOS*) de un gestor de colas que se ejecuta en IBM MQ for z/OS.

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

Multi El ejemplo siguiente muestra el objeto predeterminado proporcionado para **AUTHTYPE**(*IDPWOS*) de un gestor de colas que se ejecuta en IBM MQ for Multiplatforms.

```
 1 : display authinfo(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
AMQ8566: Display authentication information details.
   AUTHINFO(SYSTEM.DEFAULT.AUTHINFO.IDPWOS)
   AUTHTYPE(IDPWOS)                        ADOPTCTX(NO)
   DESCR( )                                CHCKCLNT(REQDADM)
   CHCKLOCL(OPTIONAL)                      FAILDLAY(1)
   ALTDATE(2015-06-08)                     ALTTIME(16.35.16)
```

AUTHINFO TYPE(IDPWOS) tiene un atributo llamado CHCKCLNT. Si el valor se cambia a *REQUIRED*, todas las aplicaciones cliente deben proporcionar un ID de usuario y una contraseña válidos.

Si el usuario se ha autenticado en el paso 7, no se volverá a autenticar a no ser que una salida de seguridad haya cambiado el usuario o la contraseña del campo SecurityParms de la estructura MQCXP en el paso 9.

Paso 11: Adoptar el contexto del usuario MQCSP (Si `ChlauthEarlyAdopt`=*N* y ADOPTCTX=YES)
Puede establecer el atributo ADOPTCTX, que controla si el canal se ejecuta bajo MCAUSER, o bajo el ID de usuario que ha proporcionado la aplicación.

Si el ID de usuario confirmado en el campo MQCSP, o **`SecurityParms`** de la estructura MQXCP, se ha autenticado correctamente y **ADOPTCTX** es *YES*, el contexto del usuario resultante de los pasos 7 y 8 se adopta como el contexto a utilizar para esta aplicación, a menos que el usuario o la contraseña en el campo **`SecurityParms`** de la estructura MQCXP se haya modificado mediante una salida de seguridad en el paso 9.

El ID de usuario certificado es el que se ha comprobado la autorización para utilizar los recursos de IBM MQ.

Por ejemplo, no tiene un MCAUSER establecido en el canal SVRCONN y el cliente se está ejecutando bajo '`johndoe`' en la máquina Linux. La aplicación especifica el usuario '`fred`' en MQCSP, así que el canal inicia la ejecución con '`johndoe`' como el MCAUSER activo. Después de la comprobación de CONNAUTH, se adopta el usuario '`fred`' y el canal se ejecuta con '`fred`' como el MCAUSER activo.

Paso 12: Comprobar que el usuario no esté bloqueado (BLOCKUSER)
Si la comprobación de **CONNAUTH** se realiza correctamente, la memoria caché de CHLAUTH se vuelve a inspeccionar para comprobar si el MCAUSER activo está bloqueado por una regla *BLOCKUSER*. Si el usuario está bloqueado, el canal finaliza.

Step13: Valide los requisitos CHLAUTH CHCKCLNT
Si la regla CHLAUTH que se ha seleccionado en el paso 8 especifica adicionalmente un valor CHCKCLNT de REQUIRED o REQDADM, se realiza la validación para asegurarse de que se ha proporcionado un ID de usuario CONNAUTH válido para cumplir el requisito.

- Si se establece CHCKCLNT (REQUIRED), un usuario debe haberse autenticado en el paso 7 o 10. De lo contrario, se rechazará la conexión.
- Si se establece CHCKCLNT (REQDADM), un usuario debe haberse autenticado en el paso 7 o 10 si se determina que esta conexión es privilegiada. De lo contrario, se rechazará la conexión.
- Si se establece CHCKCLNT (ASQMGR), se omite este paso.

Notas:

1. Si se establece CHCKCLNT (REQUIRED) o CHCKCLNT (REQDADM), pero CONNAUTH no está habilitado en el gestor de colas, la conexión falla con un código de retorno MQRC_SECURITY_ERROR (2063) debido al conflicto en la configuración.
2. El usuario no se vuelve a autenticar en este paso.

Paso 14: Validar requisitos CONNAUTH CHCKCLNT.
La fase de autenticación se produce si CONNAUTH está habilitado en el gestor de colas.

Se comprueba el valor CONNAUTH CHCKCLNT para determinar qué requisitos se establecen para las conexiones entrantes:

- Si se establece CHCKCLNT (NONE), se omite este paso
- Si se establece CHCKCLNT (OPTIONAL), este paso se omite.
- Si se establece CHCKCLNT (REQUIRED), un usuario debe haberse autenticado en el paso 7 o 10. De lo contrario, se rechazará la conexión.
- Si se establece CHCKCLNT (REQDADM), un usuario debe haberse autenticado en el paso 7 o 10 si se determina que esta conexión es privilegiada. De lo contrario, se rechazará la conexión.

Nota: El usuario no se vuelve a autenticar en este paso.

Multi **Paso 15: Comprobar autorización de objeto**
Se realiza una comprobación para asegurarse de que el MCAUSER activo tiene la autorización adecuada para conectarse al gestor de colas.

ULW Consulte Gestor de autorizaciones sobre objetos, si desea más información.

IBM i Consulte “Gestor de autorizaciones sobre objetos (OAM) en IBM i” en la página 157 para obtener más información.

Paso 16: La conexión se completa
Si los pasos precedentes se completan satisfactoriamente, la conexión se completa.

Conceptos relacionados
CONNAUTH
Un gestor de colas se puede configurar de modo que utilice un ID de usuario y contraseña proporcionados para comprobar si un usuario tiene autorización para acceder a los recursos.

Referencia relacionada
SET CHLAUTH
ALTER AUTHINFO

Resolución de problemas de acceso de CHLAUTH
Sugerencias sobre cómo resolver determinados problemas de acceso al utilizar registros de autenticación de canal (CHLAUTH).

Reglas CHLAUTH predeterminadas

Existen tres reglas predeterminadas para el proceso de CHLAUTH:

- NO ACCESS (sin acceso) en todos los canales por parte de usuarios `MQ-admin*`
- NO ACCESO a todos los SYSTEM.* canales por todos los usuarios
- ALLOW (permitir) acceso al canal SYSTEM.ADMIN.SVRCONN (usuarios no `MQ-admin`)

Las primeras dos reglas bloquean el acceso a todos los canales. La tercera regla es más específica y, por lo tanto, tiene prioridad sobre las otras dos, si el canal es el canal SYSTEM.ADMIN.SVRCONN, permitiendo así el acceso a dicho canal.

Errores comunes de conexión

Las reglas CHLAUTH se utilizan para determinar si un canal se puede iniciar, y permiten la correlación, a través de MCAUSER con otro ID de usuario. Si el canal no se puede iniciar, normalmente se producen los errores siguientes:

- RC 2035 MQRC_NOT_AUTHORIZED
- RC 2059 MQRC_Q_MGR_NOT_AVAILABLE
- AMQ4036 Acceso no permitido
- AMQ9776: El canal estaba bloqueado por el ID de usuario
- AMQ9777: El canal estaba bloqueado
- MQJE001: Se ha producido una MQException. Código de terminación 2, Razón 2035
- MQJE036: El gestor de colas ha rechazado un intento de conexión

Debería bloquear el acceso de forma estricta y, después, añadir más reglas CHLAUTH para controlar quién puede acceder e iniciar reglas. Como medida temporal, y para resolver los errores listados, puede:

- “Inhabilitar reglas CHLAUTH” en la página 62
- “Modificar o eliminar reglas CHLAUTH” en la página 62

Inhabilitar reglas CHLAUTH

Como medida temporal y, también, para resolver los errores anteriores, puede inhabilitar reglas CHLAUTH. Las reglas se pueden volver a habilitar en cualquier momento, y si la inhabilitación de las reglas CHLAUTH resuelve el problema de conexión, sabe que esta fue la causa.

Para inhabilitar reglas CHLAUTH, emita el mandato siguiente:

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

Tenga en cuenta que también puede establecer CHLAUTH en *WARN*, que permite acceder a y registrar el resultado de la regla.

Modificar o eliminar reglas CHLAUTH

También puede suprimir o modificar la regla o reglas CHLAUTH que están provocando el problema.

Para modificar una regla CHLAUTH, utilice el mandato SET CHLAUTH con ACTION (REPLACE). Por ejemplo, para modificar la regla predeterminada que provoca que no haya ningún acceso en todos los canales por parte de cualquier usuario `MQ-admin` en WARN, en lugar de su bloqueo, emita el mandato siguiente:

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

Para suprimir una regla CHLAUTH, utilice el mandato SET CHLAUTH con ACTION (REMOVE). Por ejemplo, para suprimir la regla predeterminada que provoca que no haya acceso en todos los canales por parte de los usuarios `MQ-admin`, emita el mandato siguiente:

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

Probar el acceso mediante MATCH (RUNCHECK)

Puede probar el resultado de las reglas CHLAUTH, utilizando la opción MATCH (*RUNCHECK*) de la regla CHLAUTH en runmqsc. La opción **MATCH** (*RUNCHECK*) devuelve el registro que ha comparado un canal de entrada específico durante el tiempo de ejecución, si dicho canal se conecta en este gestor de colas. Debe proporcionar:

- El nombre del canal
- Atributo ADDRESS
- Atributo SSLPEER, solo si el canal de entrada utiliza SSL o TLS
- QMNAME, si el canal de entrada es un canal de gestor de colas, o
- Atributo CLNTUSER, si el canal de entrada es un canal cliente

El ejemplo siguiente comprueba qué regla CHLAUTH, con las reglas predeterminadas en vigor, genera un usuario de `MQ-admin` `johndoe` que accede a un canal llamado `CHAN1`:

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

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

Para el usuario `johndoe`, el canal no se ejecuta, el usuario se bloqueará debido a la regla BLOCKUSER para usuarios *MQADMIN.

El ejemplo siguiente comprueba qué regla CHLAUTH, con las reglas predeterminadas en vigor, genera el usuario `alice` que no es un usuario de `MQ-admin`, que accede a un canal llamado `CHAN1`:

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

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

Para el usuario `alice`, el canal se ejecuta y el canal pasa `alice` como el MCAUSER. El MCAUSER es el ID de usuario utilizado para comprobar las autoridades sobre objeto IBM MQ.

Referencia relacionada
SET CHLAUTH
DISPLAY CHLAUTH

Creación de nuevas reglas CHLAUTH para usuarios
Algunos escenarios comunes para usuarios y reglas CHLAUTH de ejemplo para conseguirlos.

En este tema se incluyen los escenarios siguientes:

- “Control de accesos para usuarios administrativos de MQ específicos” en la página 63
- “Control de accesos para una aplicación cliente IBM MQ y un usuario específico” en la página 64
- “Control de accesos para un usuario específico utilizando el nombre distinguido (DN) del certificado de dicho usuario” en la página 64
- “Correlación de un usuario concreto con el usuario mqm” en la página 65

Control de accesos para usuarios administrativos de MQ específicos

Para este escenario, configure un canal de conexión del servidor que se vaya a utilizar de forma exclusiva para una perspectiva administrativa, es decir, para conectarse desde IBM MQ Explorer. Tiene un canal específico para este uso y la dirección o direcciones IP definidas, desde las cuales desea que se acepten conexiones, y el acceso bloqueado para el ID `'mqm'`, si la conexión no procede de una de las direcciones IP especificadas.

Cree un canal SVRCONN para los usuarios IBM MQ Explorer y `MQ-admin` llamados `ADMIN.CHAN`:

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

Para la realización de pruebas, asegúrese de que tiene un usuario definido que esté en el grupo `MQ-admin` y un usuario que no lo esté. Para este escenario, `mqadm` está en el grupo `MQ-admin` y `alice` no lo está.

Las reglas CHLAUTH predeterminadas están en vigor. Añada tres reglas para permitir que un usuario específico acceda a ADMIN.CHAN como `MQ-admin` desde determinadas direcciones IP:

- Establecer NOACCESS desde cualquier dirección
- Establecer BLOCKUSER para este canal para solo bloquear el usuario `nobody`, que altera temporalmente el *MQADMIN BLOCKUSER
- ALLOW acceso al usuario `mqadm` en una subred específica de direcciones y MAP con la autoridad de usuario `mqadm`

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

En este punto, el usuario mqadm puede acceder e iniciar el canal ADMIN.CHAN, desde el rango de direcciones IP especificado.

Puede ejecutar MATCH (RUNCHECK) en cualquier momento para ver los resultados de cada uno de estos mandatos:

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

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

En este punto, solo los usuarios que tienen un registro CHLAUTH están autorizados para acceder al uso de ADMIN.CHAN.

Control de accesos para una aplicación cliente IBM MQ y un usuario específico

Para este escenario, las reglas CHLAUTH predeterminadas son adecuadas, dando por supuesto que la autorización de IBM MQ se debe establecer para un usuario específico, para proporcionar la autorización de IBM MQ correcta (utilizando setmqaut).

En este escenario, las autorizaciones se establecen para un usuario mqapp1, que no es un usuario MQ-admin. Cree un canal SVRCONN, APP1.CHAN, que sea utilizado por una aplicación concreta y un usuario específico.

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

Con las reglas CHLAUTH predeterminadas en vigor, el usuario mqapp1 puede iniciar el canal APP1.CHAN.

El ID de usuario que procede de la aplicación cliente IBM MQ se utiliza para la comprobación de autoridades de objeto IBM MQ. En este caso, se da por supuesto que el usuario 'mqapp1' está ejecutando la aplicación cliente IBM MQ, que se utiliza para la comprobación de autoridades de objeto IBM MQ. Por lo tanto, si mqapp1 tiene acceso a los objetos IBM MQ que necesita la aplicación, todo está correcto; en caso contrario, obtendrá errores de autorización.

Puede aumentar más la seguridad creando reglas CHLAUTH específicas para el ID de usuario mqapp1, pero bajo las reglas predeterminadas, ningún miembro del grupo MQ-admin puede acceder a este canal.

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

Control de accesos para un usuario específico utilizando el nombre distinguido (DN) del certificado de dicho usuario

Para este escenario, el usuario debe tener un certificado que se haya trasladado al gestor de colas. El DN se compara con el valor SSLPEER de la regla CHLAUTH, y SSLPEER puede utilizar caracteres comodín.

Si coinciden, el usuario también se puede correlacionar con un MCAUSER diferente para finalidades de comprobación de las autoridades sobre objetos IBM MQ. La correlación de MCAUSER puede minimizar el número de usuarios que se deben gestionar en el gestor de autorizaciones sobre objetos (OAM) de IBM MQ.

Tiene un canal TLS con certificados en uso, y necesita reglas para:

- Bloquear a todos los usuarios para un canal determinado
- Permitir solo a los usuarios con un SSLPEER concreto que utilizan el cliente de dicho usuario para el acceso de OAM IBM MQ.

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

El ID de usuario del cliente que se conecta en el canal se utiliza para la autoridad de OAM IBM MQ de objetos IBM MQ; por lo tanto, el ID de usuario debe tener las autoridades de IBM MQ apropiadas.

Puede correlacionarse con un ID de usuario de IBM MQ diferente, si lo desea, utilizando:

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

en lugar de USERSRC(CHANNEL).

Correlación de un usuario concreto con el usuario mqm

Esto es una incorporación o una modificación para "Control de accesos para usuarios administrativos de MQ específicos" en la página 63.

Añada la regla CHLAUTH siguiente para correlacionar usuarios concretos con el usuario mqm, o un ID de usuario MQ-admin, que tenga la configuración de la autoridad sobre objeto IBM MQ en el OAM IBM MQ.

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

Esto habilita y correlaciona el usuario johndoe a través del usuario mqm para el canal ADMIN.CHAN concreto.

Conceptos relacionados

"Resolución de problemas de acceso de CHLAUTH" en la página 61
Sugerencias sobre cómo resolver determinados problemas de acceso al utilizar registros de autenticación de canal (CHLAUTH).

"Creación de nuevas reglas CHLAUTH para canales" en la página 65
Para ayudarle a crear sus propias reglas CHLAUTH, aquí hay algunos casos de ejemplo comunes para los canales y reglas CHLAUTH de ejemplo para llevar a cabo esto.

Referencia relacionada

SET CHLAUTH

DISPLAY CHLAUTH

Creación de nuevas reglas CHLAUTH para canales
Para ayudarle a crear sus propias reglas CHLAUTH, aquí hay algunos casos de ejemplo comunes para los canales y reglas CHLAUTH de ejemplo para llevar a cabo esto.

En este tema se incluyen los escenarios siguientes:

- "Solo permitir acceso a un canal concreto desde un rango de direcciones IP específicas." en la página 66

- “Para un canal específico, bloquear a todos los usuarios, pero permitir la conexión de usuarios específicos.” en la página 66
- “Utilización de CHLAUTH para canales receptor y emisor” en la página 66

Solo permitir acceso a un canal concreto desde un rango de direcciones IP específicas.

Para este escenario desea:

- Establecer Sin acceso en el canal desde cualquier lugar
- Permitir el acceso desde un rango de dirección o direcciones IP específicas

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

Esto permite que solo se inicie el canal APP2.CHAN cuando la conexión procede del rango de direcciones IP específicas especificado.

El usuario que se conecta como MCAUSER se correlaciona con `mqapp2` y, por lo tanto, obtiene la autorización del OAM de IBM MQ para dicho usuario.

Para un canal específico, bloquear a todos los usuarios, pero permitir la conexión de usuarios específicos.

Para este escenario, el acceso al canal MY.SVRCONN tiene las reglas CHLAUTH predeterminadas en vigor.

Tendrá que añadir lo siguiente:

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

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

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

La primera parte del código bloquea a cualquier usuario de la conexión en MY.SVRCONN, a continuación, el código solo permite que se inicie el canal MY.SVRCONN cuando la conexión procede del ID de usuario específico `johndoe`.

El usuario que se conecta en el canal `johndoe` se utiliza para la autoridad de OAM IBM MQ de los objetos IBM MQ. Por lo tanto, el ID de usuario debe tener las autorizaciones de IBM MQ apropiadas.

Puede correlacionarse con un ID de usuario de IBM MQ diferente, si lo desea, utilizando:

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

en lugar de `USERSRC(CHANNEL)`.

Utilización de CHLAUTH para canales receptor y emisor

Puede utilizar reglas CHLAUTH para añadir una seguridad adicional a los canales receptor y emisor, para restringir el acceso al canal receptor. Tenga en cuenta que, si está añadiendo o realizando cambios en reglas CHLAUTH, las reglas CHLAUTH actualizadas solo se aplican al iniciar el canal, de forma que si los canales ya se están ejecutando, tendrá que detenerlos y reiniciarlos para que se apliquen las actualizaciones de CHLAUTH.

Las reglas CHLAUTH se pueden utilizar en cualquier canal, pero existen algunas restricciones. Por ejemplo, las reglas USERMAP se aplican solo a canales SVRCONN.

Este ejemplo permite una conexión desde una dirección IP concreta solo para iniciar el canal TO.MYSVR1:

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

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

Este ejemplo solo permite la conexión desde un gestor de colas concreto:

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

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

Conceptos relacionados

“Resolución de problemas de acceso de CHLAUTH” en la página 61
Sugerencias sobre cómo resolver determinados problemas de acceso al utilizar registros de autenticación de canal (CHLAUTH).

“Creación de nuevas reglas CHLAUTH para usuarios” en la página 63
Algunos escenarios comunes para usuarios y reglas CHLAUTH de ejemplo para conseguirlos.

Referencia relacionada

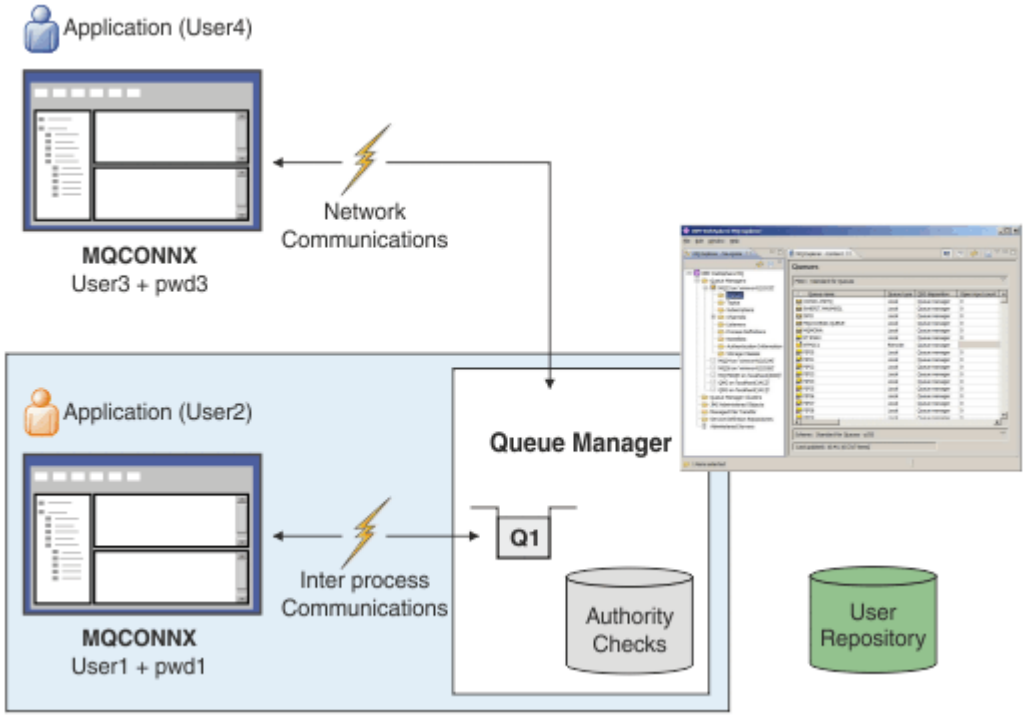
SET CHLAUTH

DISPLAY CHLAUTH

Autenticación de conexión

La autenticación de conexión se puede realizar de varias maneras:

- Una aplicación puede proporcionar un ID de usuario y una contraseña. La aplicación puede ser un cliente, o puede utilizar enlaces locales.
- Un gestor de colas se puede configurar de modo que utilice un ID de usuario y una contraseña proporcionados.
- Se puede utilizar un depósito para determinar si la combinación de ID de usuario y contraseña es válida.



En el diagrama, dos aplicaciones están realizando conexiones con un gestor de colas, una aplicación es una aplicación cliente y la otra utiliza enlaces locales. Las aplicaciones pueden utilizar diferentes API para conectarse al gestor de colas, pero todas tienen la posibilidad de proporcionar un ID de usuario y una contraseña. El ID de usuario bajo el cual se está ejecutando la aplicación, `User2` y `User4` en el diagrama, que es el ID de usuario del sistema habitual presentado en IBM MQ, podría ser diferente del ID de usuario proporcionado por la aplicación, `User1` y `User3`.

El gestor de colas recibe mandatos de configuración (en el diagrama, se utiliza IBM MQ Explorer) y gestiona la apertura de recursos y comprueba la autorización para acceder a esos recursos. Existen muchos recursos diferentes en IBM MQ para los que una aplicación podría solicitar autorización de acceso. El diagrama ilustra la apertura de una cola de salida, pero se aplican los mismos principios a otros recursos también.

Consulte la sección Depósitos de usuarios para obtener información detallada acerca del depósito que se utiliza para comprobar los ID de usuario y las contraseñas.

Conceptos relacionados

"Autenticación de conexión: Configuración" en la página 68
Un gestor de colas se puede configurar de modo que utilice un ID de usuario y contraseña proporcionados para comprobar si un usuario tiene autorización para acceder a los recursos.

"Autenticación de conexión: Cambios en la aplicación" en la página 72

"Autenticación de conexión: Depósitos de usuario" en la página 73
Para cada uno de los gestores de colas, puede seleccionar diferentes tipos de objetos de información de autenticación para autenticar los ID de usuario y las contraseñas.

Autenticación de conexión: Configuración

Un gestor de colas se puede configurar de modo que utilice un ID de usuario y contraseña proporcionados para comprobar si un usuario tiene autorización para acceder a los recursos.

Activación de la autenticación de la conexión en un gestor de colas

En un objeto de gestor de colas, el atributo **CONNAUTH** puede establecerse en el nombre de un objeto de información de autenticación (AUTHINFO). Este objeto puede ser uno de dos tipos (atributo AUTHTYPE):

IDPWOS
Indica que el gestor de colas utilizará el sistema operativo local para autenticar el ID de usuario y la contraseña.

IDPWLDAP
Indica que el gestor de colas utilizará el servidor LDAP para autenticar el ID de usuario y la contraseña.

Nota: No puede utilizar otros tipos de objetos de información de autenticación en el campo **CONNAUTH**.

IDPWOS y IDPWLDAP son similares en un número de sus atributos, que se describen aquí. Otros atributos se tendrán en cuenta más adelante.

Para comprobar las conexiones locales, utilice el atributo AUTHINFO **CHCKLOCL** (comprobar conexiones locales). Para comprobar las conexiones de cliente, utilice el atributo AUTHINFO **CHCKCLNT** (comprobar conexiones de cliente). La configuración se debe renovar para que el gestor de colas reconozca los cambios.

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

Donde USE.PW en CONNAUTH es una serie que coincide con la definición de AUTHINFO.

Tanto **CHCKLOCL** como **CHCKCLNT** tienen el mismo conjunto de valores posibles que permiten que varíe el rigor de la comprobación:

NONE
Desactiva la comprobación.

OPTIONAL
Garantiza que si una aplicación proporciona un ID de usuario y contraseña, estos deben ser un par válido, pero no es obligatorio que los proporcione. Esta opción puede resultar de utilidad durante la migración, por ejemplo.

Importante: OPTIONAL es el valor mínimo que puede establecer a fin de utilizar reglas CHLAUTH más estrictas.

Si selecciona NONE y la conexión de cliente coincide con un registro CHLAUTH con CHCKCLNT REQUIRED (o REQDADM en plataformas distintas de z/OS), la conexión falla. Recibirá el mensaje AMQ9793 en plataformas distintas de z/OS y el mensaje CSQX793E en z/OS.

REQUIRED
Requiere que todas las aplicaciones proporcionen un ID de usuario y una contraseña válidos. Consulte también la nota siguiente.

REQDADM
Los usuarios con privilegios deben suministrar un ID de usuario y contraseña válidos, pero los usuarios sin privilegios se tratan del mismo modo que con el valor OPTIONAL. Consulte también la nota siguiente. z/OS (Este valor no está permitido en sistemas z/OS).

Nota:

Si se establece **CHCKLOCL** en REQUIRED o en REQDADM significa que no puede administrar localmente el gestor de colas mediante **runmqsc** (error AMQ8135: No autorizado), a menos que el usuario especifique el parámetro -u UserId en la línea de mandatos **runmqsc**. Con esto establecido, **runmqsc** solicita la contraseña del usuario en la consola.

Del mismo modo, un usuario que ejecute IBM MQ Explorer en el sistema local verá el error AMQ4036 cuando intente conectarse al gestor de colas. Para especificar un nombre de usuario y una contraseña, pulse con el botón derecho del ratón en el objeto del gestor de colas local y seleccione **Detalles de**

conexión > **Propiedades ...** en el menú. En la sección **ID de usuario**, escriba el nombre de usuario y la contraseña que se han de utilizar, a continuación, pulse **Aceptar**.

Se aplican consideraciones similares a las conexiones remotas con **CHCKCLNT**.

CONNAUTH está en blanco para gestores de colas migrados pero establecido en *SYSTEM.DEFAULT.AUTHINFO.IDPWOS* para gestores de colas nuevos. La definición **AUTHINFO** anterior tiene **CHCKCLNT** establecida en *REQDADM* de forma predeterminada.

Por lo tanto, es necesario proporcionar la contraseña de sistema operativo correcta para todos los clientes existentes utilizando un ID de usuario privilegiado para conectarse.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de aplicación de cliente están protegidas adecuadamente, consulte “Protección por contraseña MQCSP” en la página 30.

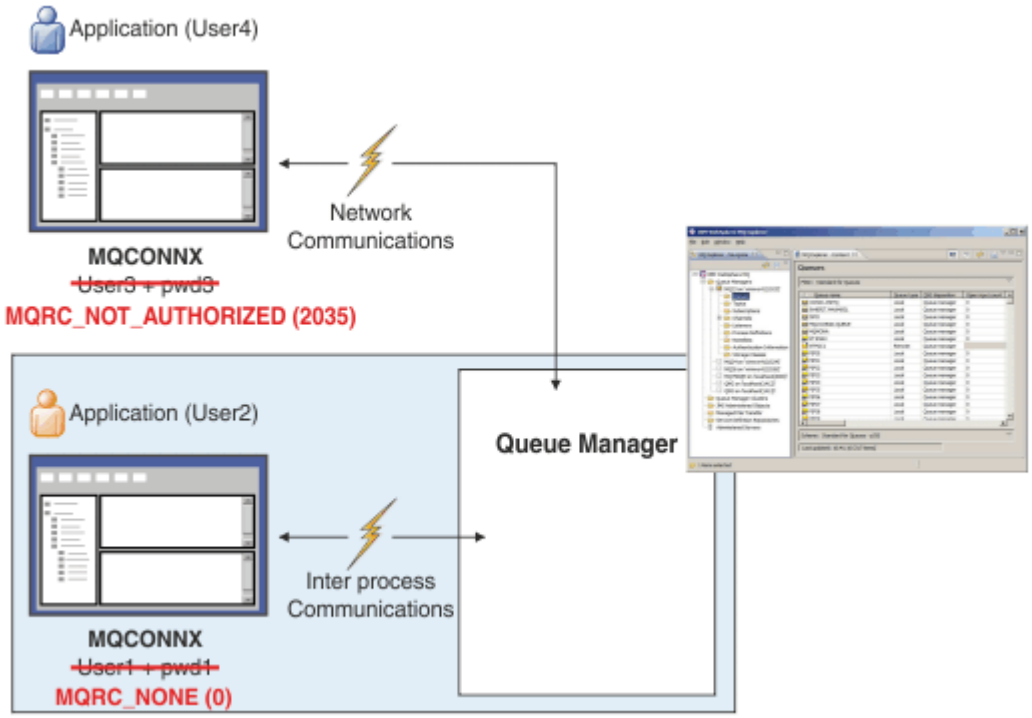
Granularidad de la configuración

Además de **CHCKLOCL** y de **CHCKCLNT** que se utilizan para activar la comprobación de ID de usuario y contraseña, existen mejoras en las reglas de CHLAUTH, de modo que puede realizarse una configuración más específicas utilizando **CHCKCLNT**.

Por ejemplo, puede establecer el valor **CHCKCLNT** global en OPTIONAL y, a continuación, actualizarlo de forma más estricta para determinados canales estableciendo **CHCKCLNT** en REQUIRED o REQDADM en la regla CHLAUTH. De forma predeterminada, las reglas CHLAUTH se ejecutarán con CHCKCLNT(ASQMGR), por lo tanto, no es necesario utilizar esta granularidad. Por ejemplo:

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

Notificación de errores



Se registra un error si una aplicación no proporciona un ID de usuario y una contraseña cuando sea necesario, o proporciona una combinación incorrecta incluso cuando es opcional.

Nota: Cuando la comprobación de contraseñas está desactivada, mediante la opción NONE en **CHCKLOCL** o **CHCKCLNT**, no se detectan las contraseñas no válidas.

Las autenticaciones con errores se conservan durante el número de segundos especificado en el atributo **FAILDLAY** antes de que el error se devuelva a la aplicación. Esto proporciona algo de protección ante una aplicación que intenta conectarse en repetidas ocasiones.

El error se graba de una serie de maneras:

Aplicación
La aplicación devuelve el error de seguridad de IBM MQ estándar, RC2035 - MQRC_NOT_AUTHORIZED.

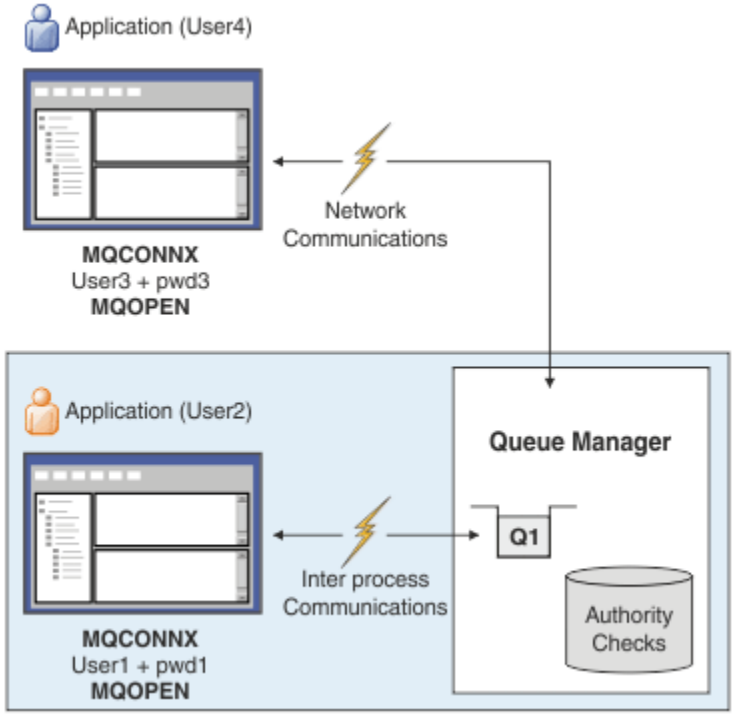
Administrador
Un administrador de IBM MQ ve el suceso notificado en el registro de errores y, por lo tanto, puede ver que la aplicación se ha rechazado porque el ID de usuario y la contraseña han fallado la comprobación, en lugar de porque, por ejemplo, no había ninguna autorización de conexión .

Herramienta de supervisión
También se puede notificar del error a una herramienta de supervisión, si activa sucesos de autorización al enviar un mensaje de suceso a la cola SYSTEM.ADMIN.QMGR.EVENT:

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

Este suceso "No autorizado" es un suceso de conexión de Tipo 1, y proporciona los mismos campos que los otros suceso de Tipo 1, con un campo adicional, el ID de usuario MQCSP que se ha proporcionado. La contraseña no se proporciona en el mensaje de sucesos. Esto significa que hay dos ID de usuario en el mensaje de suceso: el ID con el que se ejecuta la aplicación y el ID que la aplicación ha presentado para la comprobación de ID de usuario y contraseña.

Relación con la autorización



Puede configurar un gestor de colas para que requiera que se proporcionen los ID de usuario y contraseñas de determinadas aplicaciones, ya que el ID de usuario bajo el que se ejecuta la aplicación

puede no ser mismo ID de usuario que la aplicación ha presentado junto con una contraseña cuando la aplicación ha abierto la cola para salida, por ejemplo:

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

El modo en que se manejan los ID de usuario y las contraseñas se controlan mediante el atributo **ADOPTCTX** en el objeto de información de autenticación.

ADOPTCTX(YES)
Todas las comprobaciones de autorización para una aplicación se realizan con el mismo ID de usuario que ha autenticado mediante contraseña, seleccionando la adopción del contexto como contexto de la aplicación para el resto de vida de conexión.

Atención: Cuando se utiliza ADOPTCTX (YES) y los ID de usuario del sistema operativo, debe asegurarse de que el ID de usuario que se está adoptando no supera la longitud máxima de los ID de usuario. Consulte "ID de usuario" en la página 83 para obtener más información.

ADOPTCTX(NO)
Una aplicación proporciona un ID de usuario y una contraseña a efectos de autenticación en el momento de la conexión, pero luego continúa utilizando el ID de usuario con el que se ejecuta la aplicación para realizar las comprobaciones de autorización en el futuro. Es posible que encuentre esta opción útil al migrar, o si tiene previsto utilizar otros mecanismos, como registros de autenticación de canal, para asignar el identificador de usuario del agente de canal de mensajes (MCAUSER).

Atención:

Cuando utiliza el parámetro **ADOPTCTX(YES)** en un objeto de información de autenticación, no puede adoptarse otro contexto de seguridad a menos que establezca el parámetro **ChlauthEarlyAdopt** en la stanza de canales del archivo `qm.ini`.

Por ejemplo, el objeto de información de autenticación predeterminado se establece en **ADOPTCTX(YES)** y el usuario `fred` inicia una sesión. Se configuran estas dos reglas CHLAUTH:

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

Se emite el siguiente mandato, con la intención de autenticar el mandato como el contexto de seguridad adoptado del usuario bob:

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

De hecho, el gestor de colas utiliza el contexto de seguridad `fred`, no bob, y la conexión falla.

Para obtener más información sobre **ChlauthEarlyAdopt**, consulte Atributos de la stanza de canales.

Conceptos relacionados
"Autenticación de conexión" en la página 67

"Autenticación de conexión: Cambios en la aplicación" en la página 72

"Autenticación de conexión: Depósitos de usuario" en la página 73
Para cada uno de los gestores de colas, puede seleccionar diferentes tipos de objetos de información de autenticación para autenticar los ID de usuario y las contraseñas.

Autenticación de conexión: Cambios en la aplicación

Una aplicación puede proporcionar un ID de usuario y una contraseña dentro de la estructura de parámetros de seguridad de conexión (MQCSP) cuando se emite MQCONNX. Se pasan el ID de usuario y la contraseña para que los compruebe el Gestor de autorizaciones sobre objetos (OAM) que se proporciona con el gestor de colas, o el componente del servicio de autorización proporcionado con el gestor de colas en sistemas z/OS. No es necesario escribir una interfaz personalizada.

Si la aplicación se está ejecutando como un cliente, el ID de usuario y la contraseña también se pasan a las salidas de seguridad del lado de cliente y del lado del servidor para su proceso. También se pueden utilizar para establecer el atributo MCAUSER (identificador del usuario del agente de canal de mensajes) de una instancia de canal. Se invoca la salida de seguridad con la razón de salida MQXR_SEC_PARMS para este proceso. Las salidas de seguridad del lado de cliente y la salida de preconexión pueden realizar cambios en MQCONN antes de que éste se envíe al gestor de colas.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de aplicación de cliente están protegidas adecuadamente, consulte “Protección por contraseña MQCSP” en la página 30.

Si utiliza la serie XAOPEN para proporcionar un ID de usuario y una contraseña, se evitará tener que realizar cambios en el código de la aplicación.

Nota:

Desde IBM WebSphere MQ 6.0, la salida de seguridad ha permitido que se establezca MQCSP. Por lo tanto, los clientes en este nivel o posterior no tienen que actualizarse.

No obstante, en versiones de IBM MQ anteriores a la IBM MQ 8.0, MQCSP no imponía ninguna restricción en el ID de usuario y la contraseña proporcionados por la aplicación. Al utilizar estos valores con características proporcionadas por IBM MQ hay límites que se aplican a la utilización de estas características, pero si sólo las está pasando a sus propias salidas, esos límites no se aplican.

Conceptos relacionados

“Autenticación de conexión” en la página 67

“Autenticación de conexión: Configuración” en la página 68
Un gestor de colas se puede configurar de modo que utilice un ID de usuario y contraseña proporcionados para comprobar si un usuario tiene autorización para acceder a los recursos.

“Autenticación de conexión: Depósitos de usuario” en la página 73
Para cada uno de los gestores de colas, puede seleccionar diferentes tipos de objetos de información de autenticación para autenticar los ID de usuario y las contraseñas.

Autenticación de conexión: Depósitos de usuario

Para cada uno de los gestores de colas, puede seleccionar diferentes tipos de objetos de información de autenticación para autenticar los ID de usuario y las contraseñas.

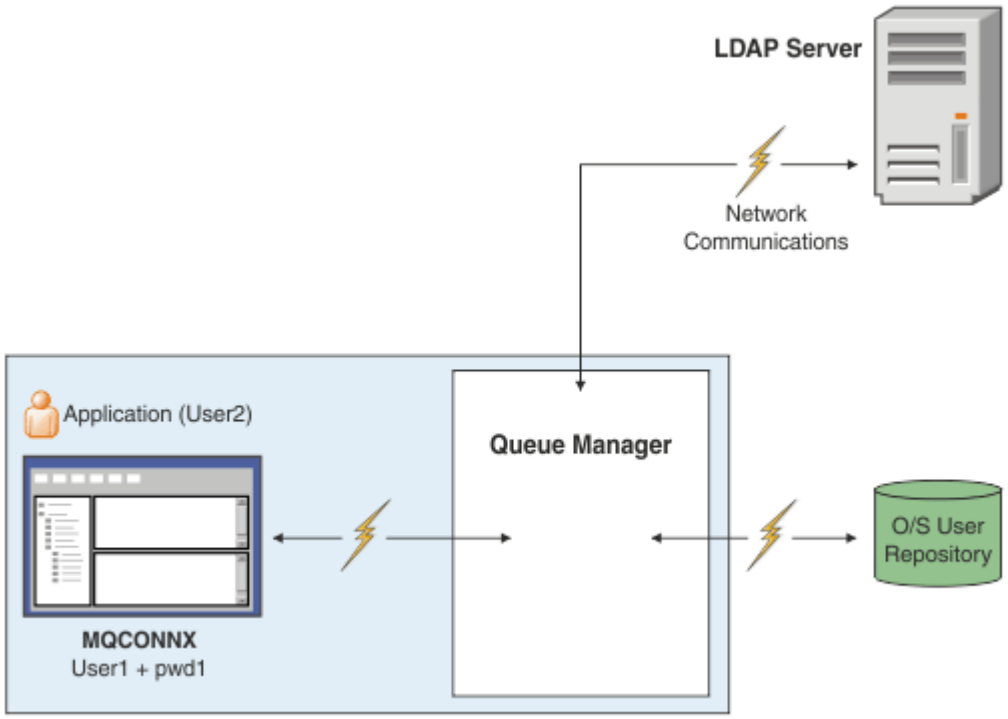


Figura 7. Tipos de objetos de información de autenticación

```
DEFINE AUTHINFO(USE.OS) AUTHTYPE(IDPWOS)
DEFINE AUTHINFO(USE.LDAP) +
AUTHTYPE(IDPWLDAP) +
CONNAME('ldap1(389),ldap2(389)') +
LDAPUSER('CN=QMGR1') +
LDAPPWD('passw0rd') SECCOMM(YES)
```

Existen dos tipos de objetos de información de autenticación, según se representa en el diagrama:

- `IDPWOS` se utiliza para indicar que el gestor de colas utiliza el sistema operativo local para autenticar el ID de usuario y la contraseña. Si opta por utilizar el sistema operativo local, necesita establecer los atributos comunes, tal como se describe en los temas anteriores.
- `IDPWLDAP` se utiliza para indicar que el gestor de colas utiliza un servidor LDAP para autenticar el ID de usuario y la contraseña. Si opta por utilizar un servidor LDAP, se proporciona más información en este tema.

Solo se puede seleccionar un objeto de información de autenticación para que lo utilice cada gestor de colas, especificando el nombre del objeto correspondiente en el atributo **CONNAUTH** del gestor de colas.

Utilización de un servidor LDAP para la autenticación.

Establezca el campo **CONNAME** en la dirección del servidor LDAP del gestor de colas. Puede proporcionar direcciones adicionales para el servidor LDAP en la lista separada por comas, lo que puede ayudarle si existen redundancias cuando el servidor LDAP no proporciona por su cuenta este recurso.

Establezca el ID de servidor y la contraseña LDAP en los campos **LDAPUSER** y **LDAPPWD**, de modo que el gestor de colas pueda acceder al servidor LDAP y buscar información acerca de los registros de usuario.

Conexión segura con un servidor LDAP

A diferencia de los canales, no existe ningún parámetro **SSLCIPH** que active el uso de TLS para la comunicación con el servidor LDAP. En este caso, IBM MQ actúa como cliente para el servidor LDAP, por lo que gran parte de la configuración se realiza en el servidor LDAP. Algunos parámetros existentes en IBM MQ se utilizan para configurar el modo en que funciona la conexión.

Establezca el campo **SECCOMM** para controlar si la conectividad con el servidor LDAP utiliza TLS.

Además de este atributo, los atributos del gestor de colas **SSLFIPS** y **SUITEB** restringen el conjunto de especificaciones de cifrado que se seleccionan. El certificado que se utiliza para identificar el gestor de colas en el servidor LDAP es el certificado del gestor de colas, ya sea `ibmwebspheremq` *`qmgr-name`* o el valor del atributo **CERTLABL** . Para obtener información detallada, consulte Etiquetas de certificados digitales.

Depósito de usuario LDAP

Si se utiliza un depósito de usuario LDAP, es necesario realizar más pasos de configuración en el gestor de colas que simplemente indicar el gestor de colas donde se encuentra el servidor AP.

Los ID de usuario definidos en un servidor LDAP tienen una estructura jerárquica que los identifica de forma exclusiva. Por lo tanto, una aplicación se puede conectar al gestor de colas y presentar su ID de usuario como el ID de usuario jerárquico totalmente calificado.

No obstante, para simplificar la información que debe proporcionar una aplicación, se puede configurar un gestor de colas que asuma que la primera parte de la jerarquía es común a todos los ID y que la añada automáticamente antes del ID abreviado que proporciona la aplicación. A continuación, el gestor de colas puede presentar un ID completo al servidor LDAP.

Establezca BASEDNU en el punto inicial en que la búsqueda LDAP busca el ID en la jerarquía de LDAP. Cuando haya establecido BASEDNU, debe asegurarse de que solo se devuelve un resultado al buscar el ID en la jerarquía de LDAP.

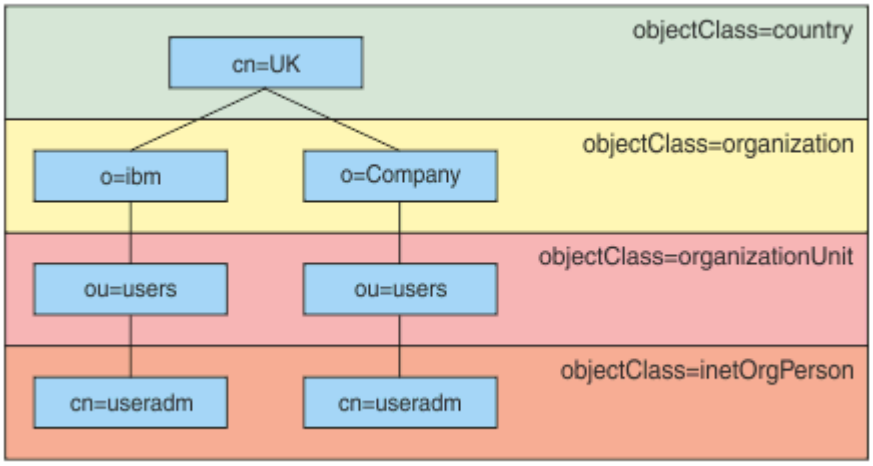


Figura 8. Un ejemplo de jerarquía de LDAP

Por ejemplo, en Figura 8 en la página 75, BASEDNU se puede establecer en "ou=users,o=ibm,c=UK" o en ",o=ibm,c=UK". No obstante, debido a que existe un nombre distinguido que contiene "cn=useradm" en la rama "o=ibm" y en la rama "o=Company", BASEDNU no se puede establecer en "c=UK". Por motivos de rendimiento y seguridad, utilice el punto más alto de la jerarquía de LDAP desde el que pueda hacer referencia a todos los ID de usuario que necesite. En este ejemplo, sería "ou=users,o=ibm,c=UK".

Su aplicación puede enviar al gestor de colas el ID de usuario sin proporcionar el nombre de atributo LDAP, por ejemplo, CN=. Si establece USRFIELD como de nombre de atributo LDAP, se añade este valor como prefijo del ID de usuario que procede de la aplicación. Esto puede resultar útil como ayuda para la migración cuando se mueven los ID de usuario del sistema operativo a los ID de usuario LDAP, ya que la aplicación puede presentar entonces la misma serie en ambos casos y evitará tener que cambiar la aplicación.

Por lo tanto, el ID de usuario completo presentado al servidor LDAP será similar al siguiente:

```
USRFIELD = ID_from_application BASEDNU
```

Conceptos relacionados

“Autenticación de conexión” en la página 67

“Autenticación de conexión: Configuración” en la página 68
Un gestor de colas se puede configurar de modo que utilice un ID de usuario y contraseña proporcionados para comprobar si un usuario tiene autorización para acceder a los recursos.

“Autenticación de conexión: Cambios en la aplicación” en la página 72

Salida de seguridad del lado del cliente para insertar ID de usuario y contraseña (mqccred)

Si tiene aplicaciones cliente que son necesarias para enviar un ID de usuario o una contraseña pero aún no puede cambiar el origen, existe una salida de seguridad que se entrega con IBM MQ 8.0 llamada **mqccred** que puede utilizar. **mqccred** proporciona un ID de usuario y una contraseña en nombre de la aplicación cliente, en un archivo `.ini`. Este ID de usuario y contraseña se envían al gestor de colas que, si se configura para ello, los autenticará.

Visión general

mqccred es una salida de seguridad que se ejecuta en la misma máquina que la aplicación cliente. Permite suministrar información de ID de usuario y contraseña en nombre de la aplicación cliente, donde dicha información no la está proporcionando la propia aplicación. La información de ID de usuario y contraseña se proporciona en una estructura conocida como Parámetros de seguridad de conexión (MQCSP) y la autenticará el gestor de colas si se ha configurado la autenticación de conexión.

La información de ID de usuario y contraseña se recupera de un archivo `.ini` en la máquina cliente. Las contraseñas del archivo se protegen mediante enmascaramiento mediante el mandato **runmqccred** y también asegurándose de que los permisos de archivo del archivo `.ini` se han establecido de modo que solo el ID de usuario que ejecuta la aplicación cliente (y por lo tanto la salida) pueda leerlo.

Ubicación

mqccred está instalado:

Plataformas Windows
En el directorio *installation_directory*\Tools\c\Samples\mqccred\

Plataformas UNIX
En el directorio *installation_directory*/samp/mqccred

Notas: La salida:

1. Funciona exclusivamente como una salida de canal de seguridad y debe ser la única salida de este tipo definida en un canal.
2. Normalmente se nombra a través de la tabla de definiciones de canal de cliente (CCDT), pero un cliente Java puede tener la salida especificada directamente en los objetos JNDI, o se puede configurar la salida para las aplicaciones que construyen manualmente la estructura MQCD.
3. Debe copiar los programas **mqccred** y **mqccred_r** en el directorio `var/mqm/exits`.

   Por ejemplo, en una máquina de la plataforma UNIX de 64 bits, emita el mandato:

   ```
   cp installation_directory/samp/mqccred/lib64/* /var/mqm/exits
   ```

   Para obtener más información, consulte Un ejemplo paso a paso de cómo probar mqccred.
4. Es capaz de ejecutarse en versiones anteriores de IBM MQ, tan lejos como IBM WebSphere MQ 7.0.1.

Configuración de ID de usuario y contraseñas

El archivo `.ini` contiene stanzas para cada gestor de colas con un valor global para gestores de colas no especificados. Cada stanza contiene el nombre del gestor de colas, un ID de usuario y una contraseña de texto sin formato o enmascarada.

Debe editar el archivo `.ini` manualmente, utilizando el editor que desee, y añada el atributo de contraseña de texto sin formato a las stanzas. Ejecute el programa **runmqccred** proporcionado, que toma el archivo `.ini` y sustituye el atributo **Password** por el atributo **OPW** , un formato ofuscado de la contraseña.

Consulte runmqccred para obtener una descripción del mandato y sus parámetros.

El archivo `mqccred.ini` contiene la información de ID de usuario y contraseña.

Se proporciona un archivo `.ini` de plantilla en el mismo directorio que la salida para proporcionar un punto de partida para la empresa.

De forma predeterminada, este archivo se buscará en `$HOME/.mqs/mqccred.ini`. Si desea ubicarlo en otro lugar, puede utilizar la variable de entorno *MQCCRED* de modo que apunte al mismo:

```
MQCCRED=C:\mydir\mqccred.ini
```

Si utiliza *MQCCRED*, la variable debe incluir el nombre completo del archivo de configuración, incluido todos los tipos de archivo `.ini`. Puesto que este archivo contiene contraseñas (incluso si están enmascaradas), se espera que se proteja el archivo utilizando privilegios de sistema operativo para asegurarse de que personas no autorizadas no puedan leerlo. Si no tiene el permiso de archivo correcto, la salida no se ejecutará satisfactoriamente.

Si la aplicación ya ha proporcionado una estructura MQCSP, la salida normalmente lo respeta y no insertará ninguna información del archivo `.ini`. No obstante, se puede alterar temporalmente utilizando el atributo **Force** en la stanza.

Si se establece **Force** en el valor *TRUE* se eliminará el ID de usuario y la contraseña proporcionados por la aplicación y se sustituirán por la versión del archivo ini.

También puede establecer el atributo **Force** en la sección global del archivo para establecer el valor predeterminado de dicho archivo.

El valor predeterminado para **Force** es *FALSE*.

Puede proporcionar un ID de usuario y contraseña para todos los gestores de colas, o para cada gestor de colas individual. A continuación se muestra un ejemplo de un archivo `mqccred.ini`:

```
# comments are permitted
AllQueueManagers:
User=abc
OPW=%^&aervrgtsr

QueueManager:
Name=QMA
User=user1
OPW=H&^dbgfh

Force=TRUE

QueueManager:
Name=QMB
User=user2
password=passw0rd
```

Notas:

1. Las definiciones de gestor de colas individuales tienen prioridad sobre el valor global.
2. Los atributos son sensibles a las mayúsculas y minúsculas.

Restricciones

Cuando esta salida está en uso, el ID de usuario local de la persona que ejecuta la aplicación no fluye del cliente al servidor. La única información de identidad disponible es desde el contenido de archivos ini.

Por lo tanto, debe configurar el gestor de colas para que utilice **ADOPTCTX(YES)**, o correlacione la solicitud de conexión de entrada con un ID de usuario adecuado a través de uno de los mecanismos disponibles, por ejemplo "Registros de autenticación de canal" en la página 49.

Importante: Si añade nuevas contraseñas, o actualiza contraseñas antiguas, el mandato **runmqccred** solo procesa todas las contraseñas de texto sin formato, dejando las enmascaradas sin modificar.

Depurar

La salida graba en el rastreo IBM MQ estándar cuando esta habilitado.

Para ayudarle a depurar los problemas de configuración, la salida también puede grabar directamente en la salida estándar.

Normalmente no se requiere ninguna configuración de dato de salida de seguridad de canal (**SCYDATA**) para el canal. Sin embargo, puede especificar:

ERROR
Imprime solo información sobre condiciones de error, como por ejemplo que no se puede encontrar el archivo de configuración.

DEBUG
Visualiza estas condiciones de error y algunas sentencias de rastreo adicionales.

NOCHECKS
Ignora las limitaciones sobre permisos de archivos y la limitación adicional que el archivo `.ini` no debe contener ninguna contraseña no protegida.

Puede poner uno o más de estos elementos en el campo **SCYDATA**, separados por comas, en cualquier orden. Por ejemplo, `SCYDATA=(NOCHECKS,DEBUG)`.

Tenga en cuenta que los elementos son sensibles a las mayúsculas y minúsculas, y deben especificarse en mayúsculas.

Utilización de `mqccred`

Una vez que ha configurado el archivo, puede invocar la salida de canal actualizando la definición de canal de conexión de cliente de modo que incluya el atributo `SCYEXIT('mqccred(ChlExit)')`:

```
DEFINE CHANNEL(channelname) CHLTYPE(clntconn) +
CONNAME(remote machine) +
QMNAME(remote qmgr) +
SCYEXIT('mqccred(ChlExit)') +
REPLACE
```

Referencia relacionada
SCYDATA
SCYEXIT
runmqccred

Autenticación de conexión con el cliente Java

La autenticación de conexión es una característica en IBM MQ que permite que el gestor de colas se configure para autenticar aplicaciones utilizando un ID de usuario y una contraseña proporcionados. Cuando la aplicación es una aplicación Java que utiliza enlaces de cliente, la autenticación de conexión puede ejecutarse en modo de compatibilidad o en modo de autenticación MQCSP.

Modalidad de compatibilidad

Antes de IBM MQ 8.0, el cliente Java podía enviar un ID de usuario y una contraseña por el canal de conexión de cliente al canal de conexión del servidor, y hacer que se suministren a una salida de seguridad en los campos **RemoteUserIdentifier** y **RemotePassword** de la estructura MQCD. En modalidad de compatibilidad, este comportamiento se retiene.

Puede utilizar esta modalidad en combinación con la autenticación de conexión y realizar la migración fuera de las salidas de seguridad que se utilizaron para realizar el mismo trabajo.

Hay que utilizar ADOPTCTX(YES) o tener otro método, por ejemplo una regla CHLAUTH basada en un certificado TLS, para establecer el MCAUSER que está en ejecución cuando se utiliza el modo de compatibilidad, ya que en este modo el ID de usuario del lado del cliente no se envía al gestor de colas.

El modo de compatibilidad de operación puede habilitarse en conexiones concretas o globalmente:

- En IBM MQ classes for Java, establezca la propiedad *MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY* a `false` en la tabla hash de propiedades que se pasa al constructor de **`com.ibm.mq.MQQueueManager`**.
- En IBM MQ classes for JMS, establezca la propiedad *JmsConstants.USER_AUTHENTICATION_MQCSP* a `false`, en la fábrica de conexiones adecuada antes de crear la conexión.
- De forma global, especifique la propiedad del sistema Java *-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication= N* en la línea de mandatos al iniciar la aplicación, tal como se muestra en el ejemplo siguiente:

```
java -Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=N application_name
```

La modalidad de compatibilidad es el valor predeterminado.

Modalidad de autenticación MQCSP

En este modo, se envía el ID de usuario del lado del cliente, así como el ID de usuario y la contraseña para autenticarse, por lo que se podrá utilizar ADOPTCTX(NO). El ID de usuario y la contraseña están disponibles a una salida de seguridad de conexión con servidor en la estructura MQCSP proporcionada en la estructura MQCXP.

Esta modalidad de operación puede habilitarse de conexión a conexión o globalmente:

- En IBM MQ classes for Java, establezca la propiedad *MQConstants.USE_MQCSP_AUTHENTICATION_PROPERTY* a `true` en la tabla hash de propiedades que se pasa al constructor de **`com.ibm.mq.MQQueueManager`**.
- En IBM MQ classes for JMS, establezca la propiedad *JmsConstants.USER_AUTHENTICATION_MQCSP* a `true`, en la correspondiente fábrica de conexiones antes de crear la conexión.
- A nivel global, establezca la propiedad de sistema *com.ibm.mq.cfg.jmqi.useMQCSPauthentication* a un valor que indique verdadero, por ejemplo, añadiendo `-Dcom.ibm.mq.cfg.jmqi.useMQCSPauthentication=Y` en la línea de mandatos.

Selección del modo de autenticación en IBM MQ Explorer

IBM MQ Explorer es una aplicación Java, de forma que estas dos modalidades, modalidad de compatibilidad y modalidad de autenticación MQCSP, también le son aplicables.

V 9.1.0 A partir de IBM MQ 9.1.0, el modo de autenticación MQCSP es el modo predeterminado. Antes de IBM MQ 9.1, el modo de compatibilidad es el predeterminado.

En paneles donde se proporciona la identificación de usuario, hay un recuadro de selección para habilitar o inhabilitar el modo de compatibilidad:

- V 9.1.0 A partir de IBM MQ 9.1.0, de forma predeterminada, esta casilla no está seleccionada de forma predeterminada. Para utilizar el modo de compatibilidad, seleccione esta casilla.
- Antes de IBM MQ 9.1.0, de forma predeterminada, esta casilla está habilitada. Para utilizar la autenticación MQCSP, desmarque la casilla.

Conceptos relacionados

"Autenticación de conexión" en la página 67

"Autenticación de conexión: Cambios en la aplicación" en la página 72

“Autenticación de conexión: Depósitos de usuario” en la página 73
Para cada uno de los gestores de colas, puede seleccionar diferentes tipos de objetos de información de autenticación para autenticar los ID de usuario y las contraseñas.

Seguridad de mensajes en IBM MQ

La seguridad de mensajes en la infraestructura de IBM MQ se proporciona por Advanced Message Security.

Advanced Message Security (AMS) amplía los servicios de seguridad de IBM MQ para proporcionar funciones de firma y cifrado de los datos a nivel de mensaje. Los servicios ampliados garantizan que los datos de los mensajes no se han modificado entre el momento en que se colocaron originalmente en una cola y cuando se recuperaron. Además, AMS verifica que el emisor de los datos de un mensaje está autorizado para colocar mensajes firmados en una cola de destino.

Conceptos relacionados

“Advanced Message Security” en la página 571
Advanced Message Security (AMS) es un componente de IBM MQ que proporciona un alto nivel de protección para los datos confidenciales que fluyen a través de la red IBM MQ, aunque no afecta a las aplicaciones finales.

Planificación de los requisitos de seguridad

En esta colección de temas se explica lo que debe tener en cuenta al planificar la seguridad en un entorno IBM MQ.

Puede utilizar IBM MQ para una amplia gama de aplicaciones de diferentes plataformas. Los requisitos de seguridad serán probablemente diferentes para cada aplicación. Para algunas, la seguridad será un tema importante.

IBM MQ proporciona una serie de servicios de seguridad a nivel de enlace, incluyendo soporte para TLS (seguridad de la capa de transporte).

Debe tener en cuenta determinados aspectos de seguridad al planificar la instalación de IBM MQ:

- Multi En Multiplatforms, si ignora estas cuestiones y no hace nada, no podrá utilizar IBM MQ.
- z/OS En z/OS, el efecto de ignorar estos aspectos es que los recursos de IBM MQ no están protegidos. Es decir, todos los usuarios pueden acceder y modificar todos los recursos de IBM MQ.

autorización para administrar IBM MQ

Los administradores de IBM MQ necesitan autorización para:

- Emitir mandatos para administrar IBM MQ
- Utilizar IBM MQ Explorer
- IBM i Utilizar los paneles y mandatos administrativos de IBM i.
- z/OS Utilizar las operaciones y los paneles de control en z/OS
- z/OS Utilizar el programa de utilidad de IBM MQ, CSQUTIL, en z/OS
- z/OS Acceder a los conjuntos de datos de gestor de colas en z/OS

Si desea ver más información, consulte:

- ULW “Autorización para administrar IBM MQ en UNIX, Linux, and Windows” en la página 409
- IBM i “Autorización para administrar IBM MQ en IBM i” en la página 85
- z/OS “Autorización para administrar IBM MQ en z/OS” en la página 86

autorización para trabajar con objetos de IBM MQ

Las aplicaciones pueden acceder a los objetos de IBM MQ siguientes emitiendo llamadas MQI:

- Gestores de colas
- Colas
- todos los Procesos
- Listas de nombres
- Temas

Las aplicaciones también pueden utilizar mandatos de Formato de mandatos programables (PCF) para acceder a estos objetos IBM MQ y para acceder también a canales y a objetos de información de autenticación. Estos objetos pueden ser protegidos por IBM MQ para que los ID de usuario asociados a las aplicaciones necesiten autorización para acceder a ellos.

Para obtener más información, consulte "Autorización para que las aplicaciones utilicen IBM MQ" en la página 88.

Seguridad de canal

Los ID de usuario asociados a los agentes de canal de mensajes (MCA) necesitan autorización para acceder a diferentes recursos de IBM MQ. Por ejemplo, un MCA debe poder conectarse a un gestor de colas. Si se trata de un MCA emisor, debe poder abrir la cola de transmisión para el canal. Si se trata de un MCA emisor, debe poder abrir las colas de destino. El ID de usuario asociados con aplicaciones que necesitan administrar canales, iniciadores de canal y escuchas necesitan autorización para utilizar los mandatos PCF pertinentes. Sin embargo, la mayoría de las aplicaciones no necesitan este tipo de acceso.

Para obtener más información, consulte "Autorización de canal" en la página 110.

Consideraciones adicionales

Debe tener en cuenta los siguientes aspectos de seguridad solamente si utiliza determinadas extensiones de funciones o de producto base de IBM MQ:

- "Seguridad para clústeres de gestores de colas" en la página 123
- "Seguridad para Publicación/Suscripción de IBM MQ" en la página 124
- "seguridad para IBM MQ Internet Pass-Thru" en la página 125

Planificación de la identificación y autenticación

Decida qué ID de usuario va a utilizar, cómo y en qué niveles desea aplicar controles de autenticación.

Debe decidir cómo va a identificar a los usuarios de las aplicaciones de IBM MQ, teniendo en cuenta que distintos sistemas operativos dan soporte a ID de usuario de longitudes diferentes. Puede utilizar registros de autenticación de canal para correlacionar de un ID de usuario a otro, o para especificar un ID de usuario basándose en algún atributo de conexión. Los canales de IBM MQ que utilizan TLS utilizan los certificados digitales como mecanismo para la identificación y autenticación. Cada certificado digital tiene un nombre distinguido de asunto que se puede correlacionar con identidades específicas utilizando registros de autenticación de canal. Además, los certificados de CA del repositorio de claves determinan qué certificados digitales se pueden utilizar para autenticar en IBM MQ. Para obtener más información, consulte:

- "Correlacionar un gestor de colas remoto con un ID de usuario MCAUSER" en la página 393
- "Correlación de un ID de usuario cliente con un ID de usuario MCAUSER" en la página 394
- "Correlacionar un Nombre distinguido SSL o TLS con un ID de usuario MCAUSER" en la página 394
- "Correlacionar una dirección IP con un ID de usuario MCAUSER" en la página 396

Planificación de la autenticación para una aplicación cliente

Puede aplicar controles de autenticación en cuatro niveles: en el nivel de comunicaciones, en las salidas de seguridad, con registros de autenticación de canal y en términos de la identificación que se ha pasado a una salida de seguridad.

Hay cuatro niveles de seguridad a tener en cuenta. El diagrama muestra un IBM MQ MQI client que está conectado a un servidor. La seguridad se aplica en cuatro niveles, tal como se describe en el texto siguiente. MCA es un Agente de canal de mensajes.

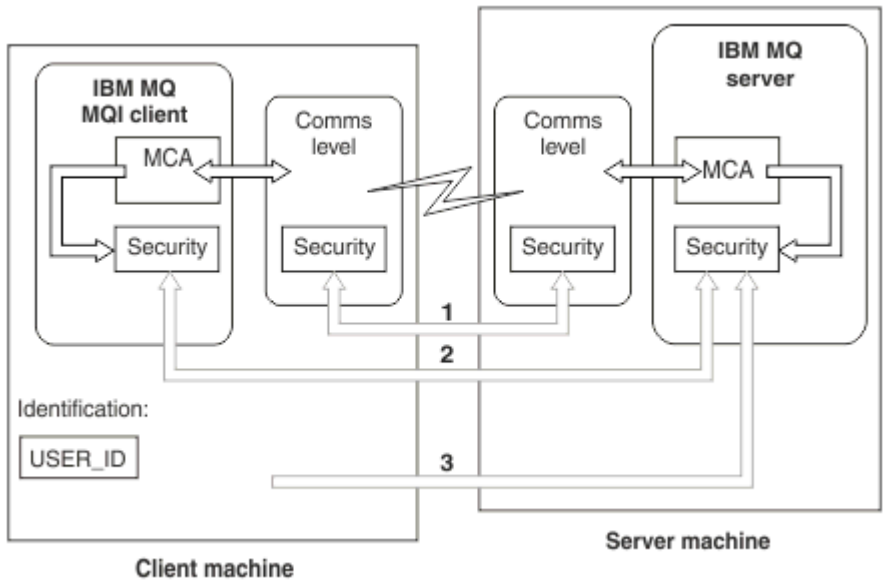


Figura 9. Seguridad en una conexión cliente/servidor

1. Nivel de comunicaciones

   Consulte la flecha 1. Para implementar la seguridad a nivel de comunicaciones, utilice TLS. Para obtener más información, consulte “Protocolos de seguridad de cifrado: TLS” en la página 15
2. Registros de autenticación de canal

   Consulte las flechas 2 y 3. La autenticación se puede controlar utilizando la dirección IP o los nombres distinguidos TLS en el nivel de seguridad. Un ID de usuario también se puede bloquear, o se puede correlacionar ID de usuario validado con un ID de usuario válido. En “Registros de autenticación de canal” en la página 49 se proporciona una descripción completa.
3. Autenticación de conexión

   Consulte la flecha 3. El cliente envía un ID y una contraseña. Para obtener más información, consulte “Autenticación de conexión: Configuración” en la página 68.
4. Salidas de seguridad de canal

   Consulte la flecha 2. Las salidas de seguridad de canal para la comunicación de cliente a servidor pueden funcionar de la misma forma que para la comunicación de servidor a servidor. Un par de salidas independientes del protocolo pueden escribirse para proporcionar la autenticación mutua tanto del cliente como del servidor. Se proporciona una descripción completa en Programas de salida de la seguridad de canal.
5. Identificación que se pasa a una salida de seguridad de canal.

   Consulte la flecha 3. En la comunicación de cliente a servidor, las salidas de seguridad de canal no tienen que funcionar como un par. La salida en el lado del cliente IBM MQ se puede omitir. En este caso, el ID de usuario se coloca en el descriptor del canal (MQCD) y la salida de seguridad del lado del servidor lo puede modificar, si es necesario.

   Los clientes de Windows también envían información adicional para ayudar a la identificación.

- El ID de usuario que se pasa al servidor es el ID de usuario que está conectado actualmente al cliente.
- El ID de seguridad del usuario conectado actualmente.

Los valores del ID de usuario y, si está disponible, el ID de seguridad, pueden ser utilizados por la salida de seguridad del servidor para establecer la identidad del IBM MQ MQI client.

A partir de IBM MQ 8.0, puede enviar contraseñas incluidas en la estructura MQCSP.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de aplicación de cliente están protegidas adecuadamente, consulte "Protección por contraseña MQCSP" en la página 30.

ID de usuario

Cuando crea ID de usuario para las aplicaciones de cliente, los ID de usuario no deben superar la longitud máxima permitida. No debe utilizar los ID de usuario reservados UNKNOWN y NOBODY. Si el servidor al que se conecta el cliente es un servidor IBM MQ for Windows , debe escapar el uso del signo de arroba, @. La longitud permitida de los ID de usuario depende de la plataforma que se utiliza para el servidor:

- z/OS Linux UNIX En z/OS y UNIX and Linux, la longitud máxima de un ID de usuario es de 12 caracteres.
- IBM i En IBM i, la longitud máxima de un ID de usuario es de 10 caracteres.
- Windows En Windows, si el IBM MQ MQI client y el servidor de IBM MQ están en Windows y el servidor tiene acceso al dominio donde se ha definido el ID de usuario del cliente, la longitud máxima de un ID de usuario es de 20 caracteres. No obstante, si el servidor de IBM MQ no es un servidor de Windows, el ID de usuario se trunca a 12 caracteres.
- Si utiliza la estructura MQCSP para pasar credenciales, la longitud máxima de un ID de usuario es de 1024 caracteres. El ID de usuario de la estructura MQCSP no se puede utilizar para eludir la longitud máxima de ID de usuario utilizada por IBM MQ para la autorización. Para obtener más información sobre la estructura MQCSP, consulte "Identificación y autenticación de usuarios utilizando la estructura MQCSP" en la página 339.

En sistemas UNIX and Linux , el valor predeterminado es que los ID de usuario se utilizan para autenticarse y los grupos se utilizan para la autorización. Sin embargo, puede configurar estos sistemas para autorizarlos en los ID de usuario. Para obtener más información, consulte "Permisos basados en usuario de OAM en UNIX and Linux" en la página 357. Los sistemas Windows pueden utilizar tanto los ID de usuario para la autenticación como para la autorización y los grupos para la autorización.

Si crea cuentas de servicio, sin prestar atención a los grupos y autoriza todos los ID de usuario de manera diferente, todos los usuarios pueden acceder a la información del resto de usuarios.

ID de usuario restringidos

Los ID de usuario `UNKNOWN` y el grupo `NOBODY` tienen significados especiales en IBM MQ. La creación de un ID de usuario en el sistema operativo denominado `UNKNOWN` o un grupo denominado `NOBODY` podría tener resultados no deseados.

Los ID de usuario cuando se conecta a un servidor de IBM MQ for Windows

Windows

Un servidor IBM MQ for Windows no da soporte a la conexión de un cliente Windows si el cliente está ejecutándose bajo un ID de usuario que contiene el carácter @, por ejemplo `abc@d`. El código de retorno para la llamada `MQCONN` en el cliente es MQRC_NOT_AUTHORIZED.

Sin embargo, puede especificar el ID de usuario utilizando dos caracteres @, por ejemplo, `abc@@d`. El uso del formato `id@domain` es la práctica preferida, para asegurarse de que el ID de usuario se resuelve en el dominio correcto de forma coherente; por lo tanto, `abc@@d@domain`.

Planificación de la autorización

Planifique los usuarios que tendrán autorización administrativa y planifique cómo autorizar a los usuarios de aplicaciones para que utilicen correctamente los objetos de IBM MQ incluidos los que se conectan desde un IBM MQ MQI client.

Para poder utilizar IBM MQ se debe otorgar acceso a personas o a aplicaciones. Qué acceso requieren dependerá de los roles que realicen y de las tareas que deban realizar. La autorización en IBM MQ puede subdividirse en dos categorías principales:

- Autorización para realizar operaciones administrativas
- Autorización para que las aplicaciones utilicen IBM MQ

Ambas clases de operación las controla el mismo componente y se puede otorgar autorización a una persona para que lleve a cabo las dos categorías de operación.

Los temas siguientes proporcionan más información sobre áreas de autorización específicas que debe tener en cuenta:

autorización para administrar IBM MQ

Los administradores de IBM MQ necesitan autorización para realizar diversas funciones. Esta autorización se obtiene de diferentes maneras en diferentes plataformas.

Los administradores de IBM MQ necesitan autorización para:

- Emitir mandatos para administrar IBM MQ.
- Windows Linux Utilizar IBM MQ Explorer.
- z/OS Utilizar las operaciones y los paneles de control en z/OS.
- z/OS Utilizar el programa de utilidad de IBM MQ, CSQUTIL, en z/OS.
- z/OS Acceder a los conjuntos de datos de gestor de colas en z/OS.

Para obtener más información, consulte el tema correspondiente a su sistema operativo.

Windows UNIX *Autorización para administrar IBM MQ en sistemas UNIX y Windows*

Un administrador de IBM MQ es un miembro del grupo mqm. Este grupo tiene acceso a todos los recursos de IBM MQ y puede emitir mandatos de control IBM MQ. Un administrador puede otorgar autorizaciones específicas a otros usuarios.

Para ser un administrador de IBM MQ en sistemas UNIX y Windows, un usuario debe ser miembro del *grupo mqm*. Este grupo se crea automáticamente cuando se instala IBM MQ. Para permitir que los usuarios emitan mandatos de control, debe añadirlos al grupo mqm. Esto incluye el usuario root UNIX.

A los usuarios que no son miembros del grupo mqm se les pueden otorgar privilegios administrativos, pero no pueden emitir mandatos de control de IBM MQ, y tienen autorización para ejecutar solamente los mandatos para los que se les ha otorgado acceso.

Además, en los sistemas Windows , las cuentas SYSTEM y Administrator tienen acceso completo a los recursos de IBM MQ .

Todos los miembros del grupo mqm tienen acceso a todos los recursos de IBM MQ del sistema, incluida la posibilidad de administrar cualquier gestor de colas que se ejecute en el sistema. Este acceso solamente se puede revocar si se suprime un usuario del grupo mqm. En los sistemas Windows, los miembros del grupo de administradores también tienen acceso a todos los recursos de IBM MQ.

Los administradores pueden utilizar el mandato **runmqsc** para emitir mandatos de script de IBM MQ (MQSC). Cuando se utiliza **runmqsc** en modalidad indirecta para enviar mandatos MQSC a un gestor de colas remoto, todo mandato MQSC se encapsula en un mandato PCF de escape. Los administradores

deben tener las autorizaciones necesarias para que el gestor de colas remoto procese los mandatos MQSC.

IBM MQ Explorer emite mandatos PCF para realizar tareas de administración. Los administradores no necesitan autorizaciones adicionales para utilizar IBM MQ Explorer para administrar un gestor de colas en el sistema local. Cuando IBM MQ Explorer se utiliza para administrar un gestor de colas en otro sistema, los administradores deben tener las autorizaciones necesarias para que el gestor de colas remoto procese los mandatos PCF.

Para obtener más información sobre las comprobaciones de autorización que se llevan a cabo cuando se procesan mandatos PCF y MQSC, consulte los temas siguientes :

- Para los mandatos que se ejecutan en gestores de colas, colas, canales, procesos, listas de nombres y objetos de información de autenticación, consulte "Autorización para que las aplicaciones utilicen IBM MQ" en la página 88.
- Para los mandatos que se ejecutan en canales, iniciadores de canal, escuchas y clústeres, consulte Seguridad de canal.
- z/OS Para los mandatos MQSC que procesa el servidor de mandatos en IBM MQ for z/OS, consulte "Seguridad de mandato y seguridad de recurso de mandato en z/OS" en la página 86.

Para obtener más información sobre la autorización que necesita para administrar IBM MQ en los sistemas UNIX y Windows, consulte la información relacionada.

IBM i *Autorización para administrar IBM MQ en IBM i*

Para ser un administrador de IBM MQ en IBM i, debe ser miembro del *grupo QMQMADM*. Este grupo tiene propiedades similares a las del grupo mqm en los sistemas UNIX y Windows. En particular, el grupo QMQMADM se crea al instalar IBM MQ for IBM i y los miembros del grupo QMQMADM tienen acceso a todos los recursos de IBM MQ en el sistema. También tiene acceso a todos los recursos IBM MQ si tiene autorización *ALLOBJ.

Los administradores pueden utilizar mandatos CL para administrar IBM MQ.Uno de estos mandatos es GRTMQMAUT, que se utiliza para conceder autorizaciones a otros usuarios. Otro mandato, STRMQMMQSC, permite que un administrador emita mandatos MQSC a un gestor de colas local.

Hay dos grupos de mandatos de CL proporcionados por IBM MQ for IBM i:

Grupo 1
Para emitir un mandato en esta categoría, un usuario debe ser miembro del grupo QMQMADM o tener autorización *ALLOBJ. Por ejemplo, GRTMQMAUT y STRMQMMQSC pertenecen a este categoría.

Grupo 2
Para emitir un mandato en esta categoría, un usuario no necesita ser miembro del grupo QMQMADM ni tener autorización *ALLOBJ. En su lugar, se necesitan dos niveles de autorización:

- El usuario necesita autorización de IBM i para utilizar el mandato. Esta autorización se concede mediante el mandato GRTOBJAUT.
- El usuario necesita autorización de IBM MQ para acceder a cualquier objeto IBM MQ asociado con el mandato. Esta autorización se concede mediante el mandato GRTMQMAUT.

Los ejemplos siguientes muestran mandatos de este grupo:

- CRTMQMQ, Crear cola MQM
- CHGMQMPRC, Cambiar proceso MQM
- DLTMQMNL, Suprimir lista de nombres MQM
- DSPMQMAUTI, Visualizar información de autenticación MQM
- CRTMQMCHL, Crear canal MQM

Para obtener más información sobre este grupo de mandatos, consulte el apartado "Autorización para que las aplicaciones utilicen IBM MQ" en la página 88.

Para obtener una lista completa de los mandatos de los grupos 1 y 2, consulte “Autorizaciones de acceso para los objetos de IBM MQ en IBM i” en la página 158

Para obtener más información sobre la autorización que necesita para administrar IBM MQ en IBM i, consulte Administración de IBM i .

z/OS *Autorización para administrar IBM MQ en z/OS*

En esta colección de temas se describen diversos aspectos de la autorización que se necesita para administrar IBM MQ for z/OS.

z/OS *comprobaciones de autorización en z/OS*

IBM MQ for z/OS utiliza SAF (System Authorization Facility) para direccionar las solicitudes de comprobaciones de autorización a un gestor de seguridad externo (ESM) como, por ejemplo, z/OS Security Server Resource Access Control Facility (RACF). IBM MQ no realiza comprobaciones de autorización propias.

Se presupone que está utilizando RACF como su ESM. Si está utilizando un ESM diferente, es posible que tenga que interpretar la información que se proporciona para RACF de un modo que sea aplicable a su ESM.

Puede especificar si desea que las comprobaciones de seguridad se activen o desactiven para cada gestor de colas individualmente o para cada uno de los gestores de colas de un grupo de compartición de colas.Este nivel de control se denomina *seguridad de subsistema*. Si desactiva la seguridad de subsistema para un gestor de colas específico, no se llevarán a cabo comprobaciones de seguridad para dicho gestor de colas.

Si activa la seguridad de subsistema para un gestor de colas específico, se pueden llevar a cabo comprobaciones de seguridad a dos niveles:

Seguridad a nivel de grupo de compartición de colas
Las comprobaciones de autorización utilizan perfiles RACF que son compartidos por todos los gestores de colas del grupo de compartición de colas. De este modo, se deberán definir y mantener menos perfiles y se facilita la administración de la seguridad.

Seguridad a nivel de gestor de colas
Las comprobaciones RACF específicos del gestor de colas.

Puede utilizar una combinación de seguridad a nivel de grupo de compartición de colas y de gestor de colas. Por ejemplo, puede organizar los perfiles específicos de un gestor de colas de modo que prevalezcan sobre los del grupo de compartición de colas, al que pertenece el gestor de colas.

La seguridad de subsistema, la seguridad a nivel de grupo de compartición de colas y la seguridad a nivel de gestor de colas se activan o desactivan definiendo *perfiles de conmutador*. Un perfil de conmutador es un perfil RACF normal que tiene un significado especial para IBM MQ.

z/OS *Seguridad de mandato y seguridad de recurso de mandato en z/OS*

La seguridad de mandatos se refiere a la autorización para emitir un mandato; la seguridad de recursos de mandatos se refiere a la autorización para realizar una operación en un recurso. Ambas se implementan utilizando clases RACF.

Las comprobaciones de autorización se llevan a cabo cuando el administrador de IBM MQ emite un mandato MQSC. Esto se denomina *seguridad de mandatos*.

Para implementar la seguridad de mandatos, debe definir ciertos perfiles RACF y otorgar a los ID de usuario y grupos necesarios acceso a estos perfiles en los niveles requeridos. El nombre de un perfil de seguridad de mandatos contiene el nombre de un mandato MQSC.

Algunos mandatos MQSC realizan una operación en un recurso IBM MQ como, por ejemplo, el mandato DEFINE QLOCAL para crear una cola local. Cuando un administrador emite un mandato MQSC, se llevan a cabo comprobaciones de autorización para determinar si la operación solicitada se puede realizar en el recurso especificado en el mandato. Esto se denomina *seguridad de recursos de mandatos*.

Para implementar la seguridad de recursos de mandatos, debe definir ciertos perfiles RACF y otorgar a los ID de usuario y grupos necesarios acceso a estos perfiles en los niveles requeridos. El nombre de un perfil de seguridad de recursos de mandatos contiene el nombre de un recurso de IBM MQ y su tipo (QUEUE, PROCESS, NAMELIST, TOPIC, AUTHINFO o CHANNEL).

La seguridad de mandatos y la seguridad de recursos de mandatos son independientes. Por ejemplo, cuando un administrador emite el mandato:

```
DEFINE QLOCAL(MOON.EUROPA)
```

se llevan a cabo las siguientes comprobaciones de autorización:

- La seguridad de mandatos comprueba que el administrador tenga autorización para emitir el mandato DEFINE QLOCAL.
- La seguridad de recursos de mandatos comprueba que el administrador tenga autorización para realizar una operación en la cola local llamada MOON.EUROPA.

La seguridad de mandatos y la seguridad de recursos de mandatos se pueden activar o desactivar definiendo perfiles de conmutador.

z/OS *Mandatos MQSC y la cola de entrada de mandatos del sistema en z/OS*

Utilice este tema para entender cómo el servidor de mandatos procesa los mandatos MQSC directamente a la cola de entrada de mandatos del sistema en z/OS.

La seguridad de mandatos y la seguridad de recursos de mandatos se utilizan también cuando el servidor recupera un mensaje que contiene un mandato MQSC de la cola de entrada de mandatos del sistema.El ID de usuario que se utiliza para las comprobaciones de autorización es el que se encuentra en el campo *UserIdentifier* del descriptor de mensaje del mensaje que contiene el mandato MQSC. Este ID de usuario debe tener las autorizaciones necesarias sobre el gestor de colas en que se procesa el mandato. Para obtener más información sobre el campo *UserIdentifier* y cómo se establece, consulte Contexto de mensaje.

Los mensajes que contienen mandatos MQSC se envían a la cola de entrada de mandatos del sistema en los casos siguientes:

- Los paneles de operaciones y los paneles de control envían los mandatos MQSC a la cola de entrada de mandatos del sistema del gestor de colas de destino. Los mandatos MQSC se corresponden con las acciones que selecciona en los paneles. El campo *UserIdentifier* de cada mensaje se establece en el ID de usuario TSO del administrador.
- La función COMMAND del programa de utilidad de IBM MQ, CSQUTIL, envía los mandatos MQSC del conjunto de datos de entrada a la cola de entrada de mandatos del sistema del gestor de colas de destino. Las funciones COPY y EMPTY envían mandatos DISPLAY QUEUE y DISPLAY STGCLASS. El campo *UserIdentifier* de cada mensaje se establece en el ID de usuario del trabajo.
- Los mandatos MQSC del conjunto de datos CSQINPX se envían a la cola de entrada de mandatos del sistema del gestor de colas con el que está conectado el iniciador del canal. El campo *UserIdentifier* de cada mensaje se establece en el ID de usuario del espacio de dirección del iniciador de canal.

  No se llevan a cabo comprobaciones de seguridad cuando se emiten mandatos MQSC desde los conjuntos de datos CSQINP1 y CSQINP2. Puede controlar quién tiene permiso para actualizar estos conjuntos de datos utilizando la protección de conjuntos de datos RACF.
- En un grupo de compartición de colas, un iniciador de canal puede enviar mandatos START CHANNEL a la cola de entrada de mandatos del sistema del gestor de colas al que está conectado. Se envía un mandato cuando se inicia un canal de salida que utiliza una cola de transmisión mediante un mecanismo de activación. El campo *UserIdentifier* de cada mensaje se establece en el ID de usuario del espacio de dirección del iniciador de canal.
- Una aplicación puede enviar mandatos MQSC a una cola de entrada de mandatos del sistema. De forma predeterminada, el campo *UserIdentifier* de cada mensaje se establece en el ID de usuario asociado a la aplicación.

- En los sistemas UNIX, Linux, and Windows, se puede utilizar el mandato de control **runmqsc** en modalidad indirecta para enviar mandatos MQSC a la cola de entrada de mandatos del sistema de un gestor de colas en z/OS. El campo *UserIdentifier* de cada mensaje se establece en el ID de usuario del administrador que ha emitido el mandato **runmqsc**.

z/OS *Acceder a los conjuntos de datos del gestor de colas en z/OS*

Los administradores de IBM MQ for z/OS necesitan autorización para acceder a los conjuntos de datos del gestor de colas. Lea este tema para conocer qué conjuntos de datos necesitan protección RACF.

Estos conjuntos de datos son:

- V 9.1.0 Los conjuntos de datos a los que se hace referencia mediante CSQINP1, CSQINP2 y CSQINPT en el procedimiento de la tarea iniciada del gestor de colas.
- Los conjuntos de páginas del gestor de colas, los conjuntos de datos de registro activo, los conjuntos de datos de registro de archivado y los conjuntos de datos de rutina de carga (BSDS)
- Los conjuntos de datos a los que hacen referencia CSQXLIB y CSQINPX en el procedimiento de tarea iniciado por el iniciador de canal

Debe proteger los conjuntos de datos para que ningún usuario que no esté autorizado pueda iniciar un gestor de colas o acceder a los datos de cualquier gestor de colas. Para ello, utilice la protección de conjunto de datos de RACF.

Autorización para que las aplicaciones utilicen IBM MQ

Cuando las aplicaciones acceden a objetos, los ID de usuario asociados a las aplicaciones necesitan la autorización adecuada.

Las aplicaciones pueden acceder a los objetos de IBM MQ siguientes emitiendo llamadas MQI:

- Gestores de colas
- Colas
- todos los Procesos
- Listas de nombres
- Temas

Las aplicaciones también pueden utilizar mandatos PCF para administrar objetos de IBM MQ. Cuando se procesa el mandato PCF, utiliza el contexto de autorización del ID de usuario que ha transferido el mensaje PCF.

En este contexto, las aplicaciones incluyen las que han escrito los usuarios y los proveedores junto con las que se proporcionan con IBM MQ for z/OS. Las aplicaciones que se proporcionan con IBM MQ for z/OS son:

- Los paneles de operaciones y los paneles de control
- El programa de utilidad de IBM MQ, CSQUTIL
- El programa de utilidad del manejador de colas de mensajes no entregados, CSQUDLQH

Las aplicaciones que utilizan IBM MQ classes for Java, IBM MQ classes for JMS, IBM MQ classes for .NET o Message Service Clients for C/C++ y .NET utilizan indirectamente la interfaz MQI.

Los MCA también emiten llamadas MQI y los ID de usuario asociados a los MCA necesitan autorización para acceder a estos objetos de IBM MQ. Para obtener más información acerca de estos ID de usuario y las autorizaciones que necesitan, consulte “Autorización de canal” en la página 110.

En z/OS, las aplicaciones también utilizan mandatos MQSC para acceder a estos objetos de IBM MQ pero la seguridad de mandatos y la seguridad de recursos de mandatos proporcionan comprobaciones de autorización en estas circunstancias. z/OS Para obtener más información, consulte “Seguridad de mandato y seguridad de recurso de mandato en z/OS” en la página 86 and “Mandatos MQSC y la cola de entrada de mandatos del sistema en z/OS” en la página 87.

En IBM i, un usuario que emite un mandato de CL del Grupo 2 podría necesitar autorización para acceder a un objeto de IBM MQ asociado al mandato. Para obtener más información, consulte "Cuándo se efectúan las comprobaciones de autorización" en la página 89.

Cuándo se efectúan las comprobaciones de autorización

Las comprobaciones de autorización se realizan cuando una aplicación intenta acceder a un gestor de colas, una cola, un proceso o una lista de nombres.

En IBM i, también se pueden realizar comprobaciones de autorización cuando un usuario emite un mandato de CL del Grupo 2 que accede a cualquiera de estos objetos de IBM MQ. Las comprobaciones se llevan a cabo en las siguientes circunstancias:

Cuando una aplicación se conecta a un gestor de colas utilizando una llamada MQCONN o MQCONNX

El gestor de colas solicita al entorno operativo el ID de usuario asociado a la aplicación. A continuación, el gestor de colas comprueba si el ID de usuario tiene autorización para conectarse al gestor de colas y retiene el ID de usuario para comprobaciones posteriores.

Los usuarios no tienen que iniciar la sesión en IBM MQ. IBM MQ presupone que los usuarios han iniciado la sesión en el sistema operativo subyacente y que éste los autentica.

Cuando una aplicación abre un objeto de IBM MQ utilizando una llamada MQOPEN o MQPUT1

Todas las comprobaciones de autorización se realizan cuando se abre un objeto y no cuando se accede al mismo posteriormente. Por ejemplo, las comprobaciones de autorización se realizan cuando una aplicación abre una cola. No se realizan cuando la aplicación coloca mensajes en la cola u los obtiene de ella.

Cuando una aplicación abre un objeto, especifica los tipos de operaciones que necesita realizar sobre el objeto. Por ejemplo, es posible que una aplicación abra una cola para explorar los mensajes que contiene y obtener los mensajes que contiene pero no para transferir mensajes a la cola. Para cada tipo de operación, el gestor de colas comprueba que el ID de usuario asociado a la aplicación tenga autorización para realizar esa operación

Cuando una aplicación abre una cola, las comprobaciones de autorización se realizan con respecto al objeto denominado en el campo `ObjectName` del descriptor de objeto. El campo `ObjectName` se utiliza en las llamadas `MQOPEN` o `MQPUT1`. Si el objeto es una cola de alias o una definición de cola remota, las comprobaciones de autorización se realizan respecto al propio objeto. No se realizan en la cola con la que se resuelven la cola de alias o la definición de la cola remota. Esto significa que el usuario no necesita tener permiso para acceder al mismo. Limite la autorización para crear colas a los usuarios con privilegios. De otro modo, los usuarios podrán eludir el control de accesos normal simplemente creando un alias.

Una aplicación puede hacer referencia a una cola remota de forma explícita. Establece los campos `ObjectName` y `ObjectQMgrName` en el descriptor de objeto en los nombres de la cola remota y el gestor de colas remoto. Las comprobaciones de autorización se realizan con respecto a la cola de transmisión con el mismo nombre que el gestor de colas remoto. En z/OS, se realiza una comprobación en el perfil de colas RACF que coincide con el nombre de gestor de colas remoto. En Multiplatforms, se realiza una comprobación en el perfil RQMNAME que coincide con el nombre de gestor de colas remoto, si se utiliza la agrupación en clúster. Una aplicación puede hacer referencia a una cola de clúster explícitamente estableciendo el campo `ObjectName` en la descripción de objeto en el nombre de la cola de clúster. Las comprobaciones de autorización se realizan sobre la cola de transmisión de clúster, `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

La autorización sobre una cola dinámica se basa en la cola modelo de la que se deriva, aunque no tiene por qué ser igual; consulte la nota 1.

El ID de usuario que el gestor de colas utiliza para las comprobaciones de autorización se obtiene del sistema operativo. El ID de usuario se obtiene cuando la aplicación se conecta al gestor de colas. Una aplicación con las autorizaciones adecuadas puede emitir una llamada `MQOPEN` especificando un ID de usuario alternativo. Las comprobaciones de control de accesos se realizan de este modo en el ID de usuario alternativo. Utilizar un ID de usuario alternativo no cambiará el ID de usuario asociado a la aplicación, solamente el que se utiliza para las comprobaciones de control de acceso.

Cuando una aplicación se suscribe a un tema utilizando una llamada MQSUB

Cuando una aplicación se suscribe a un tema, especifica el tipo de operación que necesita realizar. Está creando una suscripción o bien alterando una suscripción existente o bien reanudando una suscripción existente sin cambiarla. Para cada tipo de operación, el gestor de colas comprueba que el ID de usuario asociado a la aplicación tenga autorización para realizar la operación.

Cuando una aplicación se suscribe a un tema, las comprobaciones de autorización se realizan respecto a objetos de temas que se encuentran en el árbol de temas. Los objetos de temas están en el punto o encima del punto del árbol de temas al que se ha suscrito la aplicación. Las comprobaciones de autorización pueden implicar comprobaciones en más de un objeto de tema.El ID de usuario que el gestor de colas utiliza para las comprobaciones de autorización se obtiene del sistema operativo. El ID de usuario se obtiene cuando la aplicación se conecta al gestor de colas.

El gestor de colas realiza comprobaciones de autorización en las colas de suscriptores pero no en las colas gestionadas.

Cuando una aplicación suprime una cola dinámica persistente utilizando una llamada MQCLOSE

El manejador de objeto especificado en la llamada MQCLOSE no es necesariamente el mismo que el que ha devuelto la llamada MQOPEN que ha creado la cola dinámica persistente. Si es diferente, el gestor de colas comprueba el ID de usuario asociado a la aplicación que ha emitido la llamada MQCLOSE. Comprueba que el ID de usuario esté autorizado a suprimir la cola.

Cuando una aplicación que cierra una suscripción para eliminarla no la ha creado, se necesita la autorización de aplicación adecuada para eliminarla.

Cuando el servidor de mandatos procesa un mandato PCF que funciona en un objeto IBM MQ.

Esta regla incluye el caso en que un mandato PCF realiza una operación en un objeto de información de autenticación.

El ID de usuario que se utiliza para las comprobaciones de autorización es el que se ha encontrado en el campo `UserIdentifier` del descriptor de mensaje del mandato PCF. Este ID de usuario debe tener las autorizaciones necesarias sobre el gestor de colas en que se procesa el mandato. El mandato MQSC equivalente que se encapsula en un mandato PCF de escape se trata del mismo modo. Para obtener más información sobre el campo `UserIdentifier` y cómo establecerlo, consulte “Contexto de mensaje” en la página 91.

IBM i **En IBM i, cuando un usuario emite un mandato de CL del Grupo 2 que se ejecuta en un objeto de IBM MQ**

Esta regla incluye el caso en el que un mandato CL del Grupo 2 realiza una operación en un objeto de información de autenticación.

Las comprobaciones se realizan para determinar si el usuario tiene la autorización para realizar operaciones en un objeto de IBM MQ asociado al mandato. Las comprobaciones se realizan a menos que el usuario sea miembro del grupo QMQMADM o tenga autorización *ALLOBJ. La autorización necesaria depende del tipo de operación que el mandato realiza en el objeto. Por ejemplo, el mandato de **CHGMQMQ**, Cambiar cola MQM requiere la autorización para cambiar los atributos de la cola especificada por el mandato. Por el contrario, el mandato **DSPMQMQ**, Visualizar cola MQM requiere la autorización para visualizar los atributos de la cola especificada por el mandato.

Muchos mandatos se ejecutan en más de un objeto. Por ejemplo, para emitir el mandato **DLTMQMQ**, Suprimir cola MQM, se necesitan las autorizaciones siguientes:

- Autorización para conectar con el gestor de colas especificado en el mandato
- Autorización para suprimir la cola especificada en el mandato

Algunos mandatos no se ejecutan en ningún objeto. En este caso, el usuario sólo necesita autorización IBM i para emitir uno de estos mandatos. **STRMQMLSR** Iniciar escucha MQM, es un ejemplo de un mandato de este tipo.

Autoridad de usuario alternativo

Cuando una aplicación abre un objeto o se subscribe a un tema, la aplicación puede proporcionar un ID de usuario en la llamada MQOPEN, MQPUT1 o MQSUB. Puede solicitar al gestor de colas que utilice este ID de usuario para las comprobaciones de autorización, en lugar del ID asociado a la aplicación.

La aplicación solamente podrá abrir un objeto correctamente si se cumplen las dos condiciones siguientes:

- El ID de usuario asociado a la aplicación tiene autorización para proporcionar un ID de usuario diferente para las comprobaciones de autorización. Se considera que la aplicación tiene *autorización de usuario alternativo*.
- El ID de usuario que proporciona la aplicación tiene autorización para abrir el objeto para los tipos de operación solicitados o para suscribirse al tema.

Contexto de mensaje

La información del *contexto de mensaje* permite a la aplicación recuperar un mensaje para obtener información acerca de quién ha originado el mensaje. La información está contenida en campos del descriptor de mensaje y los campos se dividen en tres partes lógicas

Estas partes son las siguientes:

contexto de identidad
: Estos campos contienen información acerca del usuario de la aplicación que ha transferido el mensaje a la cola.

contexto de origen
: Estos campos contienen información acerca de la aplicación propiamente dicha y de cuándo se ha transferido el mensaje a la cola.

contexto de usuario
: Estos campos contienen propiedades de mensaje que las aplicaciones pueden utilizar para seleccionar mensajes que el gestor de colas debe entregar.

Cuando una aplicación transfiere un mensaje a una cola, la aplicación puede solicitar al gestor de colas que genere información de contexto en el mensaje. Esta es la acción predeterminada. Alternativamente, puede especificar que los campos de contexto no contengan información. El ID de usuario asociado a una aplicación no requiere ninguna autorización especial para realizar estas acciones.

Una aplicación puede establecer los campos de contexto de identidad en un mensaje, lo que permite que el gestor de colas genere el contexto de origen, o puede establecer todos los campos de contexto. Una aplicación también puede pasar los campos de contexto de identidad de un mensaje que ha recuperado a un mensaje que va a transferir a una cola, o puede pasar todos los campos de contexto. Sin embargo, el ID de usuario asociado a una aplicación requiere autorización para establecer o pasar información de contexto. Una aplicación especifica que desea establecer o pasar información de contexto cuando abre la cola en la que está a punto de transferir los mensajes y es en dicho momento cuando se comprueba su autorización.

La siguiente es una breve descripción de cada uno de los campos de contexto:

Contexto de identidad

UserIdentifier
: El ID de usuario asociado a la aplicación que ha transferido el mensaje. Si el gestor de colas establece este campo, se establecerá en el ID de usuario que se obtiene del sistema operativo cuando la aplicación se conecta al gestor de colas.

AccountingToken
: La información que se puede utilizar para cobrar por el trabajo realizado como resultado de un mensaje.

ApplIdentityData
: Si el ID de usuario asociado a una aplicación tiene autorización para establecer los campos de contexto de identidad o para establecer todos los campos de contexto, la aplicación puede

establecer este campo en cualquier valor relacionado con la identidad. Si el gestor de colas establece este campo, se establece en blanco.

Contexto de origen

PutApplType
El tipo de la aplicación que ha transferido el mensaje; por ejemplo, una transacción CICS.

PutApplName
El nombre de la aplicación que ha transferido el mensaje.

PutDate
La fecha en que se ha transferido el mensaje.

PutTime
La hora en la que se ha transferido el mensaje.

ApplOriginData
Si el ID de usuario asociado a una aplicación tiene autorización para establecer todos los campos de contexto, la aplicación puede establecer este campo en cualquier valor relacionado con el origen. Si el gestor de colas establece este campo, se establece en blanco.

Contexto de usuario

Los valores siguientes están soportados para **MQINQMP** o **MQSETMP**:

MQPD_USER_CONTEXT

La propiedad está asociada al contexto de usuario.

No se requiere ninguna autorización especial para poder establecer una propiedad asociada al contexto de usuario utilizando la llamada MQSETMP.

En un gestor de colas de la versión 7.0 o posterior, una propiedad asociada al contexto de usuario se guarda tal como se describe para MQOO_SAVE_ALL_CONTEXT. Una llamada MQPUT con MQOO_PASS_ALL_CONTEXT especificado hace que la propiedad se copie del contexto guardado al nuevo mensaje.

MQPD_NO_CONTEXT

La propiedad no está asociada a un contexto de mensaje.

Un valor no reconocido se rechaza con MQRC_PD_ERROR. El valor inicial de este campo es **MQPD_NO_CONTEXT**.

Para obtener una descripción detallada de cada uno de los campos de contexto, consulte MQMD - Descriptor de mensaje. Para obtener más información acerca de cómo utilizar el contexto de mensaje, consulte Contexto de mensaje.

ULW IBM i *Autorización para trabajar con objetos IBM MQ en sistemas* IBM i *IBM i , UNIX, Linux, and Windows*

El componente de servicio de autorización suministrado con IBM MQ se denomina *gestor de autorizaciones sobre objetos* (OAM). Proporciona control de acceso a través de comprobaciones de autenticación y autorización.

AUTENTICACIÓN.
La comprobación de la autenticación ejecutada por el OAM que se suministra con IBM MQ es básica y sólo se realiza en circunstancias específicas. No está diseñada para cumplir con los requisitos estrictos previstos en un entorno muy seguro.

El OAM realiza su comprobación de autenticación cuando una aplicación se conecta a un gestor de colas y se cumplen las condiciones siguientes:

- Si la aplicación de conexión ha proporcionado una estructura MQCSP, y
- Al atributo *AuthenticationType* de la estructura MQCSP se le proporciona el valor MQCSP_AUTH_USER_ID_AND_PWD y
- El valor CHCKLOCL o CHKCCLNT en el objeto AUTHINFO configurado no es 'NONE'

Los pasos de autenticación en el OAM validan la contraseña utilizando los servicios del sistema operativo, que pueden haberse configurado para realizar comprobaciones adicionales, como por ejemplo asegurarse de que el nombre de usuario no ha tenido demasiados intentos de prueba de contraseña incorrectos.

Es posible utilizar mecanismos de autenticación alternativos si escribe un nuevo componente de servicio de autorización u obtiene uno de un proveedor.

La autorización.

Las comprobaciones de la autorización son exhaustivas y no están diseñadas para cumplir la mayoría de requisitos normales.

Las comprobaciones de autorización se realizan cuando una aplicación emite una llamada MQI para acceder a un gestor de colas, una cola, un proceso o una lista de nombres. También se realizan en otros momentos; por ejemplo, cuando un mandato está siendo ejecutado por el servidor de mandatos.

En los sistemas IBM i IBM i , UNIX, Linux, and Windows, el *servicio de autorización* proporciona el control de accesos cuando una aplicación emite una llamada MQI para acceder a un objeto de IBM MQ que es un gestor de colas, cola, proceso, tema o lista de nombres. Esto incluye comprobaciones de la autorización de usuario alternativo y la autorización para establecer o pasar información de contexto.

Windows En Windows, el OAM otorga a los miembros del grupo Administradores la autorización para acceder a todos los objetos de IBM MQ, aunque el UAC esté habilitado. Además, en sistemas Windows , la cuenta SYSTEM tiene acceso completo a los recursos de IBM MQ .

El servicio de autorización también proporciona comprobaciones de autorización cuando un mandato PCF realiza operaciones en uno de estos objetos de IBM MQ o en un objeto de información de autenticación. El mandato MQSC equivalente que se encapsula en un mandato PCF de escape se trata del mismo modo.

IBM i En IBM i, a menos que el usuario sea miembro del grupo QMQMADM o tenga la autorización *ALLOBJ, el servicio de autorización también proporciona comprobaciones de autorización cuando un usuario emite un mandato de CL del Grupo 2 que se ejecuta en cualquiera de estos objetos de IBM MQ o en un objeto de información de autenticación.

El servicio de autorización es un *servicio instalable*, lo que significa que lo implementan uno o varios *componentes de servicio instalables*. Todo componente se invoca mediante una interfaz documentada. Esto permite que los usuarios y proveedores suministren componentes que mejoran o sustituyen los que se proporcionan con los productos de IBM MQ.

El componente de servicio de autorización suministrado con IBM MQ se denomina gestor de autorizaciones sobre objetos (OAM). El OAM se habilita automáticamente para cada gestor de colas que cree.

El OAM mantiene una lista de control de accesos (ACL) para cada objeto de IBM MQ al que controla el acceso. En los sistemas UNIX and Linux, sólo los ID de grupo pueden aparecer en una ACL. Esto significa que todos los miembros de un grupo tienen las mismas autorizaciones. En los sistemas IBM i IBM i y Windows, ambos ID de usuario e ID de grupo pueden aparecer en una ACL. Esto significa que se pueden otorgar autorizaciones a usuarios y grupos individuales.

Se aplica una limitación de 12 caracteres al ID de grupo y de usuario. Las plataformas UNIX suelen restringir la longitud de un ID de usuario a 12 caracteres. AIX y Linux han elevado este límite pero IBM MQ continúa cumpliendo una restricción de 12 caracteres en todas las plataformas UNIX. Si utiliza un ID de usuario de más de 12 caracteres, IBM MQ lo sustituye por el valor "UNKNOWN". No defina un ID de usuario con un valor de "UNKNOWN".

El gestor de autorizaciones sobre objetos puede autenticar un usuario y cambiar los campos de contexto de identidad apropiados. Se habilita especificando una estructura de parámetros de seguridad (MQCSP) en una llamada MQCONNX. La estructura se pasa a la función Autenticar un usuario del gestor de autorizaciones (MQZ_AUTHENTICATE_USER), que establece los campos de contexto de identidad apropiados. Si es una conexión MQCONNX desde un cliente IBM MQ, la información de MQCSP se transmite al gestor de colas al que el cliente se conecta a través de la conexión con el cliente y el canal de

conexión con el servidor. Si las salidas de seguridad se definen en dicho canal, el MQCSP se pasa a cada salida de seguridad y puede ser alterado por la salida. Las salidas de seguridad también pueden crear el MQCSP. Para obtener más detalles sobre el uso de las salidas de seguridad en este contexto, consulte Programas de salida de seguridad de canal.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de la aplicación cliente están protegidas adecuadamente, consulte IBM MQProtección de contraseña de CSP.

En los sistemas UNIX, Linux y Windows, el mandato de control **setmqaut** concede y revoca autorizaciones y se utiliza para mantener las ACL. Por ejemplo, el mandato:

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

permite que los miembros del grupo VOYAGER exploren los mensajes de la cola MOON.EUROPA propiedad del gestor de colas JUPITER. También permite que los miembros obtengan mensajes de la cola. Para revocar estas autorizaciones posteriormente, especifique el siguiente mandato:

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

El mandato:

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

permite que los miembros del grupo VOYAGER transfieran mensajes a cualquier cola cuyo nombre empiece por los caracteres MOON.. MOON.* es el nombre de un perfil genérico.Un *perfil genérico* le permite otorgar autorizaciones para un conjunto de objetos utilizando un único mandato **setmqaut** .

El mandato de control **dspmqaut** está disponible para visualizar las autorizaciones actuales que un usuario o un grupo tiene para un objeto especificado.El mandato de control **dmpmqaut** también está disponible para visualizar las autorizaciones actuales asociadas con los perfiles genéricos.

IBM i En IBM i, un administrador utiliza el mandato de CL GRTMQMAUT para otorgar autorizaciones y el mandato de CL RVKMQMAUT para revocarlas. También se pueden utilizar perfiles genéricos.Por ejemplo, el mandato de CL:

```
GRTMQMAUT MQMNAME(JUPITER) OBJTYPE(*Q) OBJ('MOON.*') USER(VOYAGER) AUT(*PUT)
```

proporciona la misma función que el ejemplo anterior de un mandato **setmqaut**; permite que los miembros del grupo VOYAGER transfieran mensajes a cualquier cola cuyo nombre empiece por los caracteres MOON.

IBM i El mandato de CL DSPMQMAUT visualiza las autorizaciones actuales que un usuario o un grupo tiene para un objeto especificado. Los mandatos de CL WRKMQMAUT y WRKMQMAUTD también están disponibles para trabajar con las autorizaciones actuales asociadas a objetos y perfiles genéricos.

Si no desea realizar comprobaciones de seguridad como sería el caso, por ejemplo, de un entorno de prueba, puede inhabilitar el OAM.

Multi *Utilización de PCF para acceder a los mandatos del gestor de autorizaciones sobre objetos (OAM)*
En sistemas IBM i, UNIX, Linux, and Windows, puede utilizar mandatos PCF para acceder a mandatos de administración de OAM.

Los mandatos PCF y los mandatos gestor de autorizaciones sobre objetos equivalentes son los siguientes:

Tabla 8. Mandatos PCF y los mandatos gestor de autorizaciones sobre objetos	
mandato PCF	**mandatos gestor de autorizaciones sobre objetos**
Consultar registros de autorización	dmpmqaut
Consultar entidad de autorización	dspmqaut
Establecer registro de autorización	setmqaut
Suprimir registro de autorización	setmqaut con -opción de eliminación

Los mandatos **setmqaut** y **dmpmqaut** están restringidos a los miembros del grupo mqm. Los mandatos PCF equivalentes los pueden ejecutar usuarios de cualquier grupo a los que se les haya otorgado autorizaciones dsp y chg en el gestor de colas.

Si desea más información sobre cómo utilizar estos mandatos, consulte Introducción a formatos de mandato programables.

z/OS *Autorización para trabajar con objetos IBM MQ en z/OS*

En z/OS, hay siete categorías de comprobación de autorización asociadas a las llamadas a la MQI. Debe definir ciertos perfiles RACF y otorgar el acceso adecuado a estos perfiles. Utilice el perfil *RESLEVEL* para controlar cuántos ID de usuario se comprueban.

Las siete categorías de comprobación de autorización asociadas a las llamadas a la MQI:

Seguridad de conexión
Las comprobaciones de autorización que se realizan cuando una aplicación se conecta a un gestor de colas.

Seguridad de colas
Las comprobaciones de autorización que se realizan cuando una aplicación abre una cola o suprime una cola dinámica permanente.

Seguridad de procesos
Las comprobaciones de autorización que se realizan cuando una aplicación abre un objeto de proceso.

Seguridad de listas de nombres
Las comprobaciones de autorización que se realizan cuando una aplicación abre un objeto de lista de nombres.

Seguridad de usuario alternativo
Las comprobaciones de autorización que se realizan cuando una aplicación solicita autorización de usuario alternativo para abrir un objeto.

Seguridad de contexto
Las comprobaciones de autorización que se realizan cuando una aplicación abre una cola y especifica que desea establecer o pasar información de contexto en los mensajes que va a transferir a la cola.

Seguridad de temas
Las comprobaciones de autorización que se realizan cuando una aplicación abre un tema.

Cada categoría de autorización se implementa del mismo modo que se implementan la seguridad de mandatos y la seguridad de recursos de mandatos. Debe definir determinados perfiles RACF y proporcionar a estos acceso a los ID de usuario y de grupo necesarios al nivel que requieran. Para la seguridad de colas, el nivel de acceso determina los tipos de operaciones que puede realizar la aplicación en una cola. Para la seguridad de contexto, el nivel de acceso determina si la aplicación puede:

- Pasar todos los campos de contexto
- Pasar todos los campos de contexto y establecer los campos de contexto de identidad
- Pasar y establecer todos los campos de contexto

Cada categoría de comprobación de autorización se puede activar o desactivar definiendo perfiles de conmutador.

Todas las categorías, excepto la seguridad de conexión, se denominan colectivamente *seguridad de recursos de la API*.

De forma predeterminada, cuando se efectúa una comprobación de seguridad de recursos de la API como resultado de una llamada MQI desde una aplicación que utiliza una conexión por lotes, solamente se comprueba un ID de usuario. Cuando se realiza una comprobación como resultado de una llamada MQI desde una aplicación CICS o IMS, o desde el iniciador de canal, se comprueban dos ID de usuario

No obstante, si define un *perfil RESLEVEL*, puede controlar si se comprueban cero, uno o dos ID de usuario. La cantidad de ID de usuario que se comprueban lo determinan el ID de usuario que se asocia al tipo de conexión cuando una aplicación se conecta al gestor de colas y el nivel de acceso que el ID de usuario tiene sobre el perfil RESLEVEL. El ID de usuario que se asocia a cada tipo de conexión es:

- El ID de usuario de la tarea de conexión para las conexiones por lotes
- El ID de usuario del espacio de direcciones del CICS para conexiones CICS
- El ID de usuario del espacio de direcciones de la región IMSpara conexiones de IMS
- El ID de usuario del espacio de direcciones del iniciador de canal para las conexiones del iniciador de canal

Para obtener más información sobre la autorización que necesita para trabajar con objetos de IBM MQ en z/OS, consulte "Autorización para administrar IBM MQ en z/OS" en la página 86.

Seguridad de la mensajería remota

En este apartado se tratan aspectos relativos a la seguridad de la mensajería remota.

Debe proporcionar autorización a los usuarios para que utilicen los recursos de IBM MQ. Esto se organiza de acuerdo con las acciones que se han de tomar con respecto a los objetos y definiciones. Por ejemplo:

- Los usuarios autorizados son los que pueden iniciar y detener los gestores de colas
- Las aplicaciones deben conectarse con el gestor de colas y tener autorización para utilizar colas
- Los usuarios autorizados deben crear y controlar los canales de mensajes
- Los objetos se mantienen en las bibliotecas y el acceso a éstas puede restringirse

El agente de canal de mensajes en un sitio remoto debe comprobar que el mensaje que se entregan se ha originado desde un usuario con autorización para hacerlo en este sitio remoto. Además, dado que los MCA se pueden iniciar de forma remota, es posible que sea necesario verificar que los procesos remotos que están intentando iniciar sus MCA estén autorizados a hacerlo. Hay cuatro posibles formas de hacerlo:

1. Utilice adecuadamente el atributo PutAuthority de la definición de canal RCVR, RQSTR o CLUSRCVR para controlar qué usuario se utiliza para las comprobaciones de autorización cuando los mensajes entrantes se colocan en las colas. Consulte la descripción del mandato DEFINE CHANNEL en la Consulta de mandatos MQSC.
2. Implemente registros de autenticación de canal para rechazar los intentos de conexión no deseados o para establecer un valor MCAUSER basada en lo siguiente: la dirección IP remota, el ID de usuario remoto, el Nombre distinguido del asunto (DN) TLS proporcionado o el nombre del gestor de colas remoto.
3. Implemente la comprobación de seguridad de la *salida de usuario* para asegurarse de que el canal de mensajes correspondiente está autorizado. La seguridad de la instalación que alberga el canal correspondiente asegura que todos los usuarios están debidamente autorizados, por lo que no es necesario comprobar los mensajes individuales.
4. Implemente el proceso de mensajes de la *salida de usuario* para asegurarse de que se comprueba la autorización de los mensajes individuales.

IBM i *Seguridad de objetos de IBM MQ for IBM i*

En este apartado se tratan aspectos relativos a la seguridad de la mensajería remota.

Debe proporcionar autorización a los usuarios para que hagan uso de los recursos de IBM MQ for IBM i. Esta autorización está organizada según las acciones que se emprenderán respecto a objetos y definiciones. Por ejemplo:

- Los usuarios autorizados son los que pueden iniciar y detener los gestores de colas
- Las aplicaciones deben conectarse al gestor de colas y tener autorización para hacer uso de las colas
- Usuarios autorizados deben crear y controlar los canales de mensajes

El agente de canal en un sitio remoto debe comprobar si el mensaje que se entrega ha derivado de un usuario con autorización para emitir el mensaje en este sitio remoto. Además, dado que los MCA se pueden iniciar de forma remota, es posible que sea necesario verificar que los procesos remotos que están intentando iniciar sus MCA estén autorizados a hacerlo. Hay cuatro posibles formas de hacerlo:

- Decrete en la definición de canal que los mensajes deben contener autorización de *contexto* aceptable; de lo contrario quedan descartados.
- Implemente registros de autorización de canal para rechazar intentos de conexión no deseados, o bien para establecer un valor MCAUSER basado en uno de los siguientes: la dirección IP remota, el ID de usuario remoto, el nombre distinguido (DN) de TLS proporcionado o el nombre del gestor de colas remoto.
- Implemente la comprobación de seguridad de salida del usuario para garantizar que el canal de mensajes correspondiente está autorizado. La seguridad de la instalación que alberga el canal correspondiente asegura que todos los usuarios están debidamente autorizados, por lo que no es necesario comprobar los mensajes individuales.
- Implemente el proceso de mensajes de salida del usuario para garantizar que los mensajes individuales se sometan a examen para su autorización.

Estos son algunos hechos sobre el modo en que IBM MQ for IBM i opera la seguridad:

- IBM i identifica y autentica a los usuarios.
- Los servicios del gestor de colas invocadas por aplicaciones se ejecutan con la autorización del perfil de usuario del gestor de colas, pero en el proceso del usuario.
- Los servicios del gestor de colas invocados mediante mandatos de usuario se ejecutan con la autorización del perfil de usuario del gestor de colas.

Linux UNIX *Seguridad de objetos en UNIX and Linux*

Los usuarios de administración deben formar parte del grupo mqm en el sistema (incluido raíz) si este ID va a utilizar mandatos de administración de IBM MQ.

Siempre debe ejecutar amqcrsta con el ID de usuario "mqm".

ID de usuario en UNIX and Linux

El gestor de colas convierte todos los identificadores de usuario en mayúsculas o en una combinación de mayúsculas/minúsculas en minúsculas. A continuación, el gestor de colas inserta los identificadores de usuario en la parte de contexto de un mensaje o comprueba su autorización. Por consiguiente, las autorizaciones sólo están basadas en identificadores en minúsculas.

Windows *Seguridad de objetos en sistemas Windows*

Los usuarios de administración deben formar parte del grupo mqm y del grupo de administradores en sistemas Windows si este ID va a utilizar mandatos de administración de IBM MQ.

ID de usuario en sistemas Windows

En sistemas Windows, *si no hay ninguna salida de mensaje instalada*, el gestor de colas convierte los identificadores en mayúsculas o en una combinación de mayúsculas/minúsculas en minúsculas. A continuación, el gestor de colas inserta los identificadores de usuario en la parte de contexto de un mensaje o comprueba su autorización. Por consiguiente, las autorizaciones sólo están basadas en identificadores en minúsculas.

ID de usuario en sistemas

En plataformas distintas a Windows, los sistemas UNIX and Linux utilizan caracteres en mayúscula para los ID de usuario en mensajes. Para permitir que los sistemas Windows, UNIX and Linux utilicen ID de usuario en minúsculas en mensajes, el agente de canal de mensajes (MCA) debe realizar las conversiones apropiadas de caracteres alfabéticos.

Para permitir que los sistemas Windows, UNIX and Linux utilicen ID de usuarios en minúsculas en mensajes, el agente de canal de mensajes (MCA) lleva a cabo las conversiones siguientes en estas plataformas:

En el extremo emisor
Los caracteres alfabéticos en todos los ID de usuario se convierten en caracteres en mayúsculas, si no hay ningún mensaje de salida instalado.

En el extremo receptor
Los caracteres alfabéticos en todos los ID de usuario se convierten en caracteres en minúsculas, si no hay ningún mensaje de salida instalado.

Las conversiones automáticas no se realizan si proporciona una salida de mensaje en UNIX, Linux, and Windows por cualquier otro motivo.

Utilización de un servicio de autorización personalizado

IBM MQ proporciona un servicio de autorización instalable. Puede optar por instalar un servicio alternativo.

El componente del servicio de autorización proporcionado con IBM MQ se llama Gestor de autoridad de objeto (OAM). Si el OAM no proporciona los recursos de autorización que necesita, puede escribir su propio componente de servicio de autorización. Las funciones de servicio instalables que debe implementar un componente de servicio de autorización se describen en Información de consulta de la interfaz de servicios instalables.

Control de accesos para clientes

El control de accesos se basa en los ID de usuario. Puede haber muchos ID de usuario para administrar, y pueden estar en distintos formatos. Puede establecer la propiedad de canal de conexión del servidor MCAUSER en un valor de ID de usuario especial para que puedan utilizarla los clientes.

El control de acceso en IBM MQ está basado en los ID de usuario. Normalmente se utiliza ID de usuario del proceso que realiza llamadas MQI. En el caso de clientes MQI de MQ, el MCA de conexión con el servidor efectúa llamadas MQI en nombre de clientes MQI de MQ. Puede seleccionar un ID de usuario alternativo para que lo utilice el MCA de conexión con el servidor para efectuar llamadas MQI. El ID de usuario alternativo se puede asociar con la estación de trabajo del cliente o lo que elija para organizar y controlar el acceso de los clientes. El ID de usuario debe tener las autorizaciones necesarias asignadas a éste en el servidor para emitir llamadas MQI. Elegir un ID de usuario alternativo es preferible a permitir que los clientes efectúen llamadas MQI con la autorización del MCA de conexión con el cliente.

Tabla 9. El ID de usuario utilizado por un canal de conexión del servidor

ID de usuario	Cuándo se utiliza
ID de usuario establecido por una salida de seguridad	Se utiliza a menos que lo bloquee una regla **CHLAUTH TYPE(BLOCKUSER)**. Consulte la sección siguiente, “Establecimiento del ID de usuario en una salida de seguridad” en la página 99, si desea más información.
ID de usuario establecido por una regla CHLAUTH	Se utiliza a menos que una salida de seguridad lo sobrescriba. Consulte Registros de autenticación de canal para obtener más información.
ID de usuario definido en el atributo **MCAUSER** en la definición de canal SVRCONN	Se utiliza a menos que una salida de seguridad o una regla CHLAUTH lo sobrescriban.

Tabla 9. El ID de usuario utilizado por un canal de conexión del servidor (continuación)	
ID de usuario	**Cuándo se utiliza**
ID de usuario que fluye desde la máquina cliente	Se utiliza cuando no se haya establecido ningún ID de usuario de cualquier otro modo.
ID de usuario que ha iniciado el canal de conexión del servidor	Se utiliza si no se ha establecido ningún ID de usuario de ningún otro modo y no se ha producido un flujo de ID de usuario de cliente. Consulte la sección siguiente, "El ID de usuario que ejecuta el programa de canal" en la página 100, si desea más información.

Puesto que el MCA de conexión con el servidor efectúa llamadas MQI en nombre de usuarios remotos, es importante tener en cuenta las implicaciones de seguridad del MCA de conexión con el servidor que emite llamadas MQI en nombre de clientes remotos y cómo administrar el acceso de un gran número potencial de usuarios.

- Una alternativa es que el MCA de conexión con el servidor emita llamadas MQI con su propia autorización. Pero tenga en cuenta que, no es deseable generalmente que el MCA de conexión con el servidor, con sus potentes prestaciones de acceso, emita llamadas MQI en nombre de usuarios de cliente.
- Otra alternativa es utilizar el ID de usuario que fluye del cliente. El MCA de conexión con el servidor emite llamadas MQI utilizando las prestaciones de acceso del ID de usuario del cliente. Este enfoque presenta una serie de preguntas que hay que tener en cuenta:
  1. Existen diferentes formatos para el ID de usuario en diferentes plataformas. Esto a veces provoca problemas si el formato del ID de usuario en el cliente difiere de los formatos aceptables en el servidor.
  2. Existen potencialmente muchos clientes, con ID de usuario diferentes y cambiantes. Los ID deben definirse y gestionarse en el servidor.
  3. ¿Es el ID de usuario fiable? Cualquier ID de usuario puede transmitirse de un cliente, no necesariamente el ID del usuario registrado. Por ejemplo, el cliente puede transmitir un ID con plena autorización `mqm` que intencionadamente sólo estaba definido en el servidor por razones de seguridad.
- La alternativa preferida es definir las señales de identificación del cliente en el servidor y limitar así las posibilidades de las aplicaciones conectadas del cliente. Esto se suele realizar estableciendo la propiedad de canal de conexión con el servidor MCAUSER en un valor de ID de usuario especial que utilizarán los clientes y definiendo unos pocos ID para que los utilicen los clientes con diferente nivel de autorización en el servidor.

Establecimiento del ID de usuario en una salida de seguridad

Para IBM MQ MQI clients, el proceso que emite las llamadas MQI es el MCA de conexión con el servidor. El ID de usuario que utiliza el MCA de conexión con el servidor está contenido en los campos `MCAUserIdentifier` o `LongMCAUserIdentifier` del MQCD. El contenido de estos campos viene establecido por:

- Cualquier valor definido por las rutinas de salida de seguridad
- El ID de usuario del cliente
- MCAUSER (en la definición de canal de conexión con el servidor)

La salida de seguridad puede prevalecer sobre los valores que son visibles, cuando se invoca.

- Si el atributo MCAUSER del canal de conexión con el servidor no está establecido en blanco, se utiliza el valor MCAUSER.
- Si el atributo MCAUSER del canal de conexión con el servidor está en blanco, se utiliza el ID de usuario procedente del cliente.

- Si el atributo MCAUSER del canal de conexión con el cliente está en blanco y no se recibe ningún ID de usuario del cliente, se utiliza el ID de usuario que inició el canal de conexión con el servidor.

El cliente IBM MQ no fluye el ID de usuario declarado hasta el servidor cuando se está utilizando una salida de seguridad del lado del cliente.

El ID de usuario que ejecuta el programa de canal

Cuando los campos de ID de usuario se derivan del ID de usuario que inició el canal de conexión con el servidor, se utiliza el valor siguiente:

- z/OS Para z/OS, el ID de usuario asignado a la tarea iniciada de iniciador de canal mediante la tabla de procedimientos iniciados de z/OS.
- Para TCP/IP (no z/OS), el ID de usuario de la entrada `inetd.conf` o el ID de usuario que ha iniciado el escucha.
- Para SNA (no z/OS), el ID de usuario de la entrada de servidor SNA o (si no hay ninguno) la solicitud de conexión entrante o el ID de usuario que ha iniciado el escucha.
- Para NetBIOS o SPX, el ID de usuario que ha iniciado el escucha.

Si existe alguna definición de canal de conexión con el servidor cuyo atributo MCAUSER esté en blanco, los clientes pueden utilizar esa definición de canal para conectarse con el gestor de colas con una autorización de acceso determinada por el ID de usuario suministrado por el cliente. Esto puede representar un riesgo para la seguridad si el sistema en el que se ejecuta el gestor de colas permite conexiones no autorizadas a la red. El canal de conexión de servidor predeterminado de IBM MQ (SYSTEM.DEF.SVRCONN) tiene el atributo MCAUSER establecido en blanco. Para impedir el acceso no autorizado, actualice el atributo MCAUSER de la definición predeterminada con un ID de usuario que no tenga acceso a los objetos de IBM MQ MQ.

El caso de los ID de usuarios

Cuando defina un canal con `runmqsc`, el atributo MCAUSER quedará en mayúsculas a menos que el ID de usuario esté entre comillas simples.

ULW En servidores de UNIX, Linux, and Windows, el contenido del campo `MCAUserIdentifier` que se recibe del cliente cambia a minúsculas.

IBM i En servidores de IBM i, el contenido del campo `LongMCAUserIdentifier` que se recibe del cliente cambia a mayúsculas.

Linux UNIX En servidores en sistemas UNIX and Linux, el contenido del campo `LongMCAUserIdentifier` que se recibe del cliente cambia a minúsculas.

De forma predeterminada, el ID de usuario que se pasa cuando se utiliza una aplicación de enlace IBM MQ JMS, es el ID de usuario para la JVM en la que se está ejecutando la aplicación.

También es posible pasar un ID de usuario a través del método `createQueueConnection`.

Planificación de la confidencialidad

Debe planificar mantener los datos confidenciales.

Puede implementar la confidencialidad a nivel de aplicación o a nivel de enlace. Puede elegir utilizar TLS, en cuyo caso debe planificar el uso de certificados digitales. También puede utilizar programas de salida de canal si los recursos estándares no satisfacen los requisitos.

Conceptos relacionados

“Comparación entre la seguridad a nivel de enlace y la seguridad a nivel de aplicación” en la página 101
Este tema contiene información sobre distintos aspectos de la seguridad de nivel de enlace y de nivel de aplicación y compara los dos niveles de seguridad.

“Programas de salida de canal” en la página 106

Los *programas de salida de canal* son programas a los que se llama en lugares definidos de la secuencia de proceso de un MCA. Los usuarios y proveedores pueden escribir sus propios programas de salida de canal. IBM proporciona algunos de ellos.

"Protección de canales con SSL/TLS" en la página 113
El soporte de TLS en IBM MQ utiliza el objeto de información de autenticación del gestor de colas y diversos mandatos MQSC. También debe contemplar el uso de certificados digitales.

Comparación entre la seguridad a nivel de enlace y la seguridad a nivel de aplicación

Este tema contiene información sobre distintos aspectos de la seguridad de nivel de enlace y de nivel de aplicación y compara los dos niveles de seguridad.

La seguridad a nivel de enlace y la seguridad a nivel de aplicación se ilustran en la Figura 10 en la página 101.

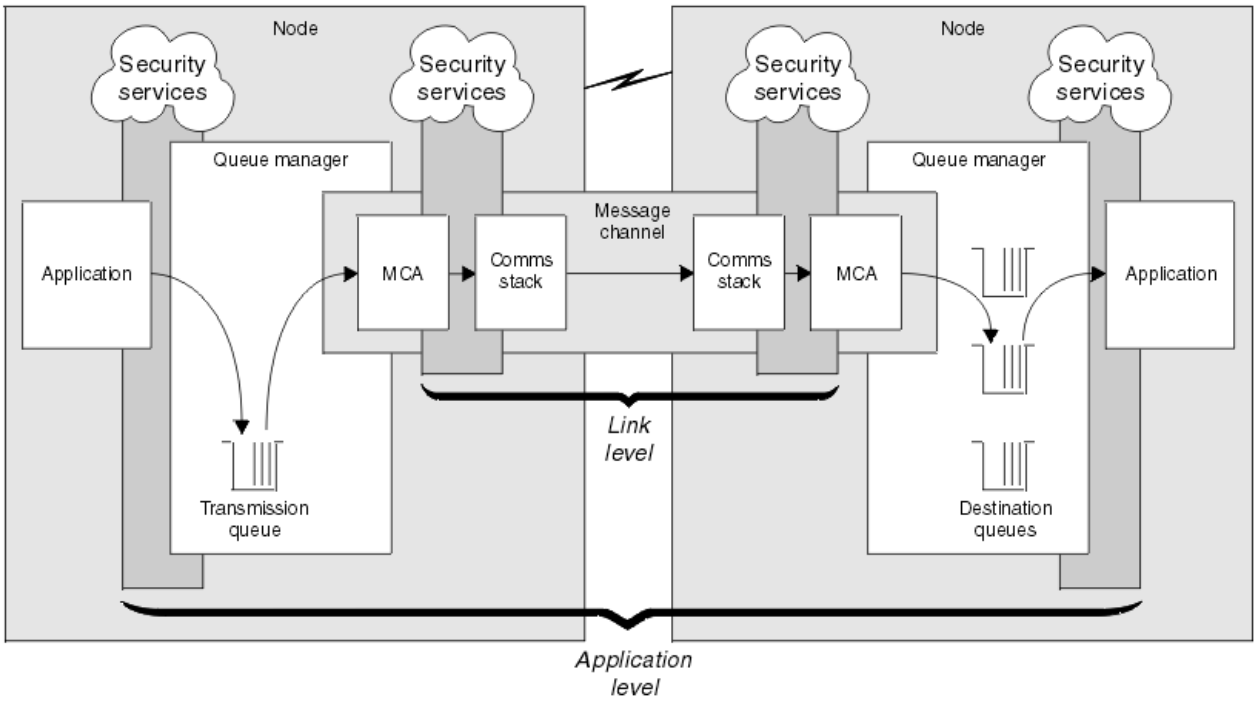


Figura 10. Seguridad a nivel de enlace y seguridad a nivel de aplicación

Protección de los mensajes en las colas

La seguridad a nivel de enlace puede proteger los mensajes mientras se transfieren de un gestor de colas a otro. Esto es especialmente importante cuando los mensajes se transmiten a través de una red que no es segura. No obstante, no puede proteger los mensajes mientras están almacenados en las colas de un gestor de colas de origen, de un gestor de colas de destino o de un gestor de colas intermedio.

z/OS V 9.1.4 El cifrado de conjuntos de datos de z/OS puede proporcionar cierta protección de los mensajes almacenados en las colas, pero sólo para los datos que se encuentran en reposo en un gestor de colas local. Consulte la sección Confidencialidad para los datos en reposo en IBM MQ for z/OS con cifrado de conjunto de datos. para obtener más información.

La seguridad a nivel de aplicación puede, en comparación, proteger los mensajes cuando están almacenados en colas y se aplica incluso cuando no se utiliza la gestión de colas distribuida. Esta es la diferencia más importante entre la seguridad a nivel de enlace y la seguridad a nivel de aplicación que se ilustra en la Figura 10 en la página 101.

Gestores de colas que no se ejecutan en entornos controlados y fiables

Si un gestor de colas se está ejecutando en un entorno controlado y de confianza, los mecanismos de control de accesos que proporciona IBM MQ pueden considerarse suficientes para proteger los mensajes almacenados en las colas. Esto es especialmente cierto si solamente se trata de gestionar colas locales y los mensajes no salen nunca del gestor de colas. La seguridad a nivel de aplicación en este caso puede considerarse innecesaria.

La seguridad a nivel de aplicación también puede considerarse innecesaria si los mensajes se transfieren a otro gestor de colas que también está ejecutándose en un entorno controlado y fiable o si se reciben desde un gestor de colas de este tipo. La necesidad de seguridad a nivel de aplicación aumenta cuando los mensajes se transfieren a o se reciben de un gestor de colas que no está ejecutándose en un entorno controlado y fiable.

Diferencias de coste

La seguridad a nivel de aplicación puede costar más que la seguridad a nivel de enlace en lo que se refiere a la administración y el rendimiento.

Es probable que el coste de la administración sea mayor porque potencialmente hay más restricciones a la hora de configurar y mantener. Por ejemplo, es posible que deba asegurarse de que un usuario determinado envía solamente determinados tipos de mensajes y envía mensajes solamente a determinados destinos. Por el contrario, es posible que tenga que asegurarse de que un usuario determinado recibe solamente determinados tipos de mensajes y únicamente de determinadas fuentes. En lugar de gestionar los servicios de seguridad a nivel de enlace en un solo canal de mensajes, es posible que tenga que configurar y mantener reglas para cada par de usuarios que intercambie mensajes a través de dicho canal.

El rendimiento puede verse afectado si los servicios de seguridad se invocan cada vez que una aplicación transfiere u obtiene un mensaje.

Las organizaciones tienden a considerar en primer lugar la seguridad a nivel de enlace porque resulta más fácil de implementar. La seguridad a nivel de aplicación la tienen en cuenta si descubren que la seguridad a nivel de enlace no satisface todos sus requisitos.

Disponibilidad de los componentes

Generalmente, en un entorno distribuido, un servicio de seguridad requiere un componente en al menos dos sistemas. Por ejemplo, es posible que un mensaje se cifre en un sistema y se descifre en otro. Esto se aplica a la seguridad a nivel de enlace y a la seguridad a nivel de aplicación.

En un entorno heterogéneo en el que se utilicen diferentes plataformas y cada una de las cuales tenga diferentes niveles de funciones de seguridad, es posible que los componentes necesarios de un servicio de seguridad no estén disponibles para cada plataforma en la que se necesitan y con un formato que resulte fácil de utilizar. Probablemente, esto se deba tener más en cuenta en la seguridad a nivel de aplicación que en la seguridad a nivel de enlace, sobre todo si piensa proporcionar su propio nivel de seguridad a nivel de aplicación comprando componentes de fuentes diferentes.

Mensajes de la cola de mensajes no entregados

Si un mensaje está protegido por la seguridad a nivel de aplicación, es posible que exista algún problema si, por algún motivo, el mensaje no llega a su destino y se coloca en una cola de mensajes no entregados. Si no encuentra el modo de procesar el mensaje a partir de la información incluida en el descriptor de mensaje y la cabecera de la cola de mensajes no entregados, es posible que tenga que examinar el contenido de los datos de la aplicación. Esto no podrá hacerlo si los datos de la aplicación están cifrados y el único que puede descifrarlos es el destinatario.

Funciones que no puede realizar la seguridad a nivel de aplicación

La seguridad a nivel de aplicación no es una solución completa. Incluso si implementa la seguridad a nivel de aplicación, es posible que necesite algunos servicios de seguridad a nivel de enlace. Por ejemplo:

- Cuando se inicia un canal, la autenticación mutua de los dos MCA puede seguir siendo un requisito. Esto solamente puede llevarlo a cabo mediante el servicio de seguridad a nivel de enlace.
- La seguridad a nivel de aplicación no puede proteger la cabecera de la cola de transmisión, MQXQH, que incluye el descriptor de mensaje intercalado. Ni tampoco puede proteger los datos de los flujos de protocolo de canal de IBM MQ que no sean los datos de mensaje. Solamente la seguridad de enlace puede proporcionar esta protección.
- Si se invocan los servicios de seguridad a nivel de aplicación en el extremo del servidor de un canal MQI, los servicios no pueden proteger los parámetros de las llamadas MQI que se envían a través del canal. En especial, los datos de la aplicación de una llamada MQPUT, MQPUT1 o MQGET no están protegidos. Solamente la seguridad a nivel de enlace puede proporcionar protección en este caso.

Seguridad a nivel de enlace

La *seguridad a nivel de enlace* hace referencia a los servicios de seguridad que invoca, de forma directa o indirecta, un MCA, el subsistema de comunicaciones o una combinación de ambos que funcionen conjuntamente.

La seguridad a nivel de enlace se ilustra en la Figura 10 en la página 101.

Los siguientes son ejemplos de servicios de seguridad a nivel de enlace:

- El MCA a cada extremo de un canal de mensajes puede autenticar a su asociado. Esto se lleva a cabo cuando se inicia el canal y se establece una conexión de comunicaciones pero antes de que se inicie el flujo de los mensajes. Si la autenticación no se ejecuta correctamente en alguno de los extremos, el canal se cierra y no se transfiere ningún mensaje. Este es un ejemplo de un servicio de identificación y autenticación.
- Se puede cifrar un mensaje en el extremo emisor de un canal y descifrar en el extremo receptor. Este es un ejemplo de un servicio de confidencialidad.
- Un mensaje se puede comprobar en el extremo receptor de un canal para determinar si el contenido se ha modificado de forma deliberada mientras se estaba transmitiendo a través de la red. Este es un ejemplo de un servicio de integridad de datos.

Seguridad a nivel de enlace proporcionada por IBM MQ

El principal medio de provisión de confidencialidad e integridad de datos en IBM MQ es mediante el uso de TLS. Para obtener más información sobre el uso de TLS en IBM MQ, consulte “Protocolos de seguridad TLS en IBM MQ” en la página 24. Para la autenticación, IBM MQ proporciona el recurso para utilizar registros de autenticación de canal. Los registros de autenticación de canal ofrecen un control preciso sobre el acceso otorgado a los sistemas que se conectan, a nivel de canales individuales o de grupos de canales. Para obtener más información, consulte “Registros de autenticación de canal” en la página 49.

Cómo proporcionar su propia seguridad a nivel de enlace

Puede proporcionar sus propios servicios de seguridad de nivel de enlace. Escribir sus propios programas de salida de canal es el método principal para proporcionar sus propios servicios de seguridad a nivel de enlace.

Se proporciona una introducción a los programas de salida de canal en “Programas de salida de canal” en la página 106. El mismo tema también describe el programa de salida de canal que se proporciona con IBM MQ for Windows (el programa de salida de canal SSPI). Este programa de salida de canal se suministra en formato fuente para que pueda modificar el código fuente para ajustarlo a sus necesidades. Si este programa de salida de canal, o los programas de salida de canal disponibles de otros proveedores, no se ajustan a sus requisitos, puede diseñar y escribir el suyo propio. En este tema se sugieren formas en que los programas de salida de canal pueden proporcionar servicios de seguridad. Para obtener más información sobre cómo escribir un programa de salida de canal, consulte Escritura de programas de salida de canal.

Seguridad a nivel de enlace mediante una salida de seguridad

Las salidas de seguridad suelen funcionar en pares: una en cada extremo de un canal. Se les llama inmediatamente después de que la negociación inicial de datos se ha completado en el inicio del canal.

Se pueden utilizar salidas de seguridad para proporcionar identificación y autenticación, control de accesos y confidencialidad.

Seguridad a nivel de enlace mediante una salida de mensajes
Una salida de mensajes sólo se puede utilizar en un canal de mensajes, no en un canal MQI. Tiene acceso tanto a la cabecera de colas de transmisión, MQXQH, que incluye el descriptor de mensaje incorporado, como a los datos de aplicación de un mensaje. Puede modificar el contenido del mensaje y cambiar su longitud.

Se puede utilizar una salida de mensajes para cualquier finalidad que requiera acceso al mensaje completo, más que a una parte del mismo.

Se pueden utilizar salidas de mensajes para proporcionar identificación y autenticación, control de accesos, confidencialidad, integridad de datos y servicio contra rechazos, y por motivos que no sean la seguridad.

Seguridad a nivel de enlace mediante salidas de emisión y recepción
Las salidas de emisión y recepción se pueden utilizar tanto en canales de mensajes como en canales MQI. Se les llama para todos los tipos de datos que fluyen en un canal y para flujos en ambas direcciones.

Las salidas de emisión y recepción tienen acceso a cada segmento de transmisión. Pueden modificar su contenido y cambiar su longitud.

En un canal de mensajes, si un MCA tiene que dividir un mensaje y enviarlo en más de un segmento de transmisión, se llama a una salida de emisión para cada segmento de transmisión que contiene una parte del mensaje y, en el extremo receptor, se llama a una salida de recepción para cada segmento de transmisión. Lo mismo sucede en un canal MQI si los parámetros de entrada o de salida de una llamada MQI son demasiado grandes como para que se envíen en un solo segmento de transmisión.

En un canal MQI, el byte 10 de un segmento de transmisión identifica la llamada MQI e indica si el segmento de transmisión contiene los parámetros de entrada o de salida de la llamada. Las salidas de emisión y recepción examinan este byte para determinar si la llamada MQI contiene datos de aplicación que se deban proteger.

Cuando se llama a una salida de emisión por primera vez, para adquirir e inicializar los recursos que necesita, puede solicitar al MCA que reserve una cantidad especificada de espacio en el almacenamiento intermedio que contiene un segmento de transmisión. Cuando se le llama posteriormente para procesar un segmento de transmisión, puede utilizar este espacio para añadir una clave cifrada o una firma digital, por ejemplo. La salida de recepción correspondiente en el otro extremo del canal puede eliminar los datos añadidos por la salida de emisión y utilizarlos para procesar el segmento de transmisión.

Las salidas de emisión y recepción son las más adecuadas en los casos en que no es necesario que comprendan la estructura de los datos que están manejando y, por lo tanto, pueden tratar cada segmento de transmisión como un objeto binario.

Se pueden utilizar salidas de emisión y recepción para proporcionar confidencialidad e integridad de datos, y por motivos que no sean la seguridad.

Tareas relacionadas
Identificación de la llamada API en un programa de salidas de envío o recepción

Seguridad a nivel de aplicación
La *seguridad a nivel de aplicación* hace referencia a los servicios de seguridad que se invocan en la interfaz entre una aplicación y un gestor de colas al que está conectada.

Estos servicios se invocan cuando la aplicación emite llamadas MQI dirigidas al gestor de colas. Los servicios los puede invocar, directa o indirectamente, la aplicación, el gestor de colas, otro producto que dé soporte a IBM MQ, o una combinación de cualquiera de esos productos que funcionen conjuntamente. La seguridad a nivel de aplicación se ilustra en la Figura 10 en la página 101.

La seguridad a nivel de aplicación se conoce también como *seguridad de extremo a extremo* o *seguridad a nivel de mensaje*.

Los siguientes son ejemplos de servicios de seguridad a nivel de aplicación:

- Cuando una aplicación transfiere un mensaje a una cola, el descriptor de mensaje contiene un ID de usuario asociado a la aplicación. No obstante, no hay datos presentes como, por ejemplo, una contraseña cifrada, que se puedan utilizar para autenticar el ID de usuario. Un servicio de seguridad puede añadir estos datos. Cuando la aplicación receptora recupera el mensaje, otro componente del servicio puede autenticar el ID de usuario utilizando los datos que ha transportado el mensaje. Este es un ejemplo de un servicio de identificación y autenticación.
- Un mensaje se puede cifrar cuando una aplicación lo transfiere a una cola y se puede descifrar cuando la aplicación receptora lo recupera.Este es un ejemplo de un servicio de confidencialidad.
- Un mensaje se puede comprobar cuando la aplicación receptora lo recupera. Esta comprobación determina si el contenido se ha modificado de forma deliberada ya que, en primer lugar, la aplicación emisora lo había transferido a la cola.Este es un ejemplo de un servicio de integridad de datos.

Planificación de Advanced Message Security
Advanced Message Security (AMS) es un componente de IBM MQ que proporciona un alto nivel de protección para los datos confidenciales que fluyen a través de la red IBM MQ, aunque no afecta a las aplicaciones finales.

Si está moviendo información delicada o valiosa, especialmente información confidencial o relacionada con los pagos como por ejemplo registros de pacientes o detalles de tarjetas de crédito, debe poner una atención especial en la seguridad de la información. Asegurarse de que la información que mueve la empresa conserva su integridad y está protegida frente al acceso no autorizado es un reto y una responsabilidad actual. También deberá cumplir con las regulaciones de seguridad, pudiendo sufrir sanciones en caso de no cumplirlas.

Puede desarrollar sus propias extensiones de seguridad para IBM MQ. Sin embargo, tales soluciones requieren habilidades especiales y pueden ser complicadas y caras de mantener. Advanced Message Security le ayuda a enfrentarse a estos retos al mover información dentro de la empresa entre virtualmente cualquier tipo de sistema de tecnologías de la información comercial.

Advanced Message Security amplía las características de seguridad de IBM MQ de las maneras siguientes:

- Proporciona protección de datos de principio a fin en el nivel de aplicación para su infraestructura de mensajes punto a punto, utilizando el cifrado o la firma digital de mensajes.
- Proporciona una seguridad exhaustiva sin escribir código de seguridad complejo ni modificar ni volver a compilar aplicaciones existentes.
- Utiliza la tecnología de Infraestructura de claves públicas (PKI) para proporcionar servicios de autenticación, autorización, confidencialidad e integridad de datos para mensajes.
- Proporciona la administración de políticas de seguridad para servidores distribuidos y de sistema principal.
- Soporta los servidores y los clientes de IBM MQ.
- Se integra con Managed File Transfer para proporcionar una solución de mensajería segura de principio a fin.

Para obtener más información, consulte “Advanced Message Security” en la página 571.

Cómo proporcionar su propia seguridad a nivel de aplicación
Puede proporcionar sus propios servicios de seguridad de nivel de aplicación. Para ayudarle a implementar la seguridad a nivel de aplicación, IBM MQ proporciona dos salidas, la salida de API y la salida cruzada de API.

La salida de API y la salida cruzada de API pueden proporcionar identificación y autenticación, control de accesos, confidencialidad, integridad de datos y servicios de no repudiación y otras funciones no relacionadas con la seguridad.

Si la salida de API o la salida cruzada de API no están soportadas en su entorno de sistema, es posible que desee considerar otros modos de proporcionar su propia seguridad a nivel de aplicación. Un método es desarrollar una API de nivel superior que encapsule la MQI. A continuación, los programadores utilizan esta API, en lugar de la MQI, para escribir aplicaciones IBM MQ.

Los motivos más comunes para utilizar una API de nivel superior son:

- Ocultar las funciones más avanzadas de la MQI a los programadores.
- Aplicar los estándares que utiliza la MQI.
- Añadir funciones a la MQI. Esta función adicional puede ser servicios de seguridad.

Algunos productos de proveedores utilizan esta técnica para proporcionar seguridad a nivel de aplicación para IBM MQ.

Si piensa proporcionar servicios de seguridad de este modo, tenga en cuenta lo siguiente en relación con la conversión de datos:

- Si se ha añadido una señal de seguridad, como por ejemplo una firma digital, a los datos de la aplicación contenidos en un mensaje, cualquier código que efectúe la conversión de datos deberá tener en cuenta la existencia de esta señal.
- Una señal de seguridad puede haberse derivado de una imagen binaria de los datos de aplicación. Por lo tanto, cualquier comprobación de la señal se debe realizar antes de convertir los datos.
- Si los datos de aplicación que contiene un mensaje se han cifrado, se deben descifrar antes de la conversión de datos.

Programas de salida de canal

Los *programas de salida de canal* son programas a los que se llama en lugares definidos de la secuencia de proceso de un MCA. Los usuarios y proveedores pueden escribir sus propios programas de salida de canal. IBM proporciona algunos de ellos.

Hay varios tipos de programas de salida de canal, pero sólo cuatro ofrecen seguridad a nivel de enlace:

- Salida de seguridad
- Salida de mensajes
- Salida de emisión
- Salida de recepción

Estos cuatro tipos de programa de salida de canal se ilustran en la Figura 11 en la página 106 y se describen en los temas siguientes.

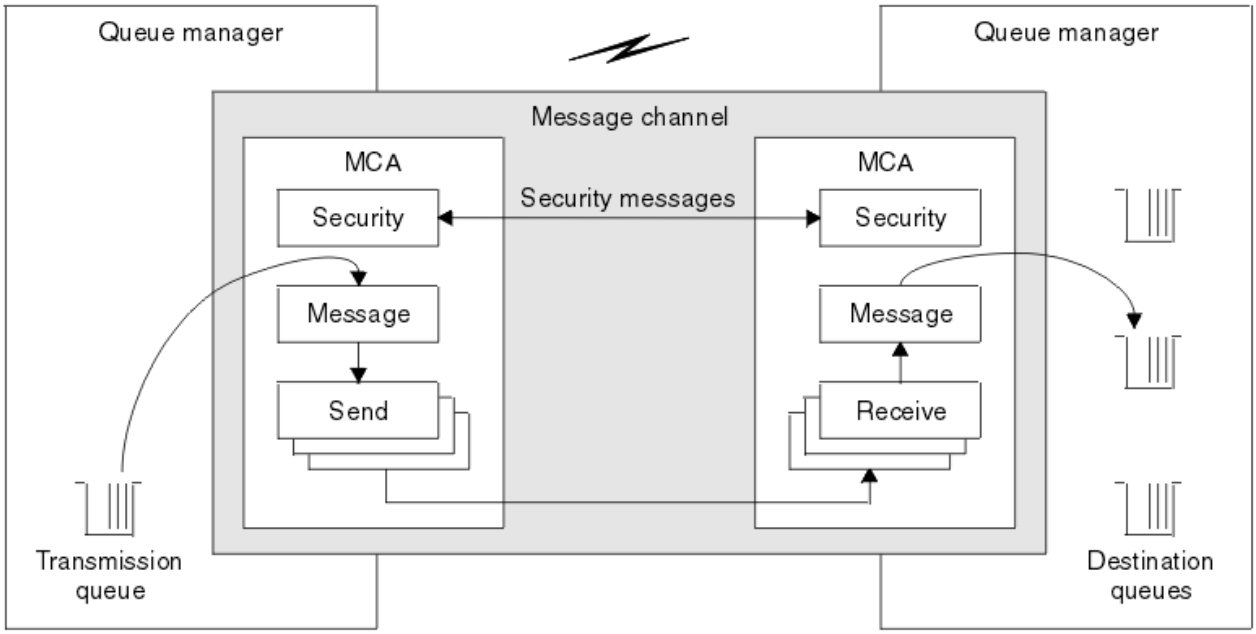


Figura 11. Salidas de seguridad, mensajes, emisión y recepción en un canal de mensajes

Conceptos relacionados

Programas de salida de canal para canales de mensajes

Visión general de las salidas de seguridad

Las salidas de seguridad normalmente funcionan en pares. Se les llama antes de que se inicie el flujo de mensajes y su finalidad es permitir que un MCA autentique su asociado.

Las *salidas de seguridad* suelen funcionar en pares; una en cada extremo de un canal. Se les llama inmediatamente después de que la negociación inicial de datos se ha completado en el inicio del canal, pero antes de que los mensajes empiecen a fluir. El principal objetivo de la salida de seguridad es permitir que el MCA de cada extremo de un canal autentique su asociado. Sin embargo, no existe ningún método para evitar que una salida de seguridad lleve a cabo otra función, incluso funciones que no tienen nada que ver con la seguridad.

Las salidas de seguridad se pueden comunicar entre sí enviando *mensajes de seguridad*. El formato de un mensaje de seguridad no está definido y lo determina el usuario. Un posible resultado del intercambio de mensajes de seguridad es que una de las salidas de seguridad decida no continuar. En este caso, el canal se cierra y los mensajes no fluyen. Si hay una salida de seguridad en un solo extremo de un canal, se sigue llamando a la salida y esta puede decidir entre continuar o cerrar el canal.

Se puede llamar a las salidas de seguridad en canales de mensajes y de MQI. El nombre de una salida de seguridad se especifica como un parámetro en la definición de canal en cada extremo del canal.

Para obtener más información acerca de las salidas de seguridad, consulte la publicación “Seguridad a nivel de enlace mediante una salida de seguridad” en la página 103.

Salida de mensajes

Las salidas de mensajes solamente funcionan en canales de mensajes y normalmente funcionan en pares. Una salida de mensajes puede funcionar en todo el mensaje y realizar diversos cambios en el mismo.

Las *salidas de mensajes* en los extremos emisor y receptor de un canal suelen funcionar en pares. Se llama a una salida de mensajes en el extremo emisor de un canal después de que el MCA haya obtenido el mensaje de la cola de transmisión. En el extremo receptor de un canal, se llama a una salida de mensajes antes de que el MCA coloque un mensaje en su cola de destino.

Una salida de mensajes tiene acceso tanto a la cabecera de colas de transmisión, MQXQH, que incluye el descriptor de mensaje incorporado, como a los datos de aplicación en un mensaje. Una salida de mensajes puede modificar el contenido del mensaje y cambiar su longitud. Un cambio en la longitud puede dar lugar a la compresión, descompresión, cifrado o descifrado del mensaje. También puede dar lugar a la adición de datos al mensaje o a la eliminación de datos del mismo.

Las salidas de mensajes se pueden utilizar para cualquier objetivo que requiera acceso al mensaje completo, no a una parte del mismo, y no necesariamente por motivos de seguridad.

Una salida de mensajes puede determinar que el mensaje que está procesando actualmente no debe continuar hacia su destino. Luego el MCA transfiere el mensaje a la cola de mensajes no entregados. Una salida de mensajes también puede cerrar el canal.

Sólo se puede llamar a salidas de mensajes en canales de mensajes, no en canales MQI. Esto se debe a que el objetivo de un canal MQI es permitir que los parámetros de entrada y salida de las llamadas MQI fluyan entre la aplicación IBM MQ MQI client y el gestor de colas.

El nombre de una salida de mensajes se especifica como un parámetro de la definición de canal en cada extremo del canal. También puede especificar una lista de salidas de mensajes para que se ejecuten en sucesión.

Para obtener más información acerca de las salidas de mensajes, consulte la publicación “Seguridad a nivel de enlace mediante una salida de mensajes” en la página 104.

Salidas de emisión y recepción

Las salidas de emisión y recepción normalmente funcionan en pares. Actúan en segmentos de transmisión y es mejor utilizarlas cuando la estructura de los datos que están procesando no es relevante.

Una *salida de emisión* en un extremo de un canal y una *salida de recepción* en el otro extremo suelen funcionar en pares. Se llama a una salida de emisión justo antes de que un MCA emita un envío de comunicaciones para enviar datos a través de la conexión de comunicaciones. Se llama a una salida de recepción justo después de que un MCA haya vuelto a obtener el control que sigue a una recepción de comunicaciones y haya recibido datos de una conexión de comunicaciones. Si se utiliza la compartición de conversaciones, a través de un canal MQI, para cada conversación se llama a una instancia distinta de una salida de envío y recepción.

Los flujos del protocolo de canal de IBM MQ entre dos MCA en un canal de mensajes contienen información de control y datos del mensaje. De forma similar, en un canal MQI, los flujos contienen información de control, así como los parámetros de llamadas MQI. Se llama a salidas de emisión y recepción para todos los tipos de datos.

Los datos del mensaje fluyen en una sola dirección en un canal de mensajes pero, en un canal MQI, los parámetros de entrada de una llamada MQI fluyen en una dirección y los parámetros de salida fluyen en la otra. Tanto en los canales de mensajes como en los MQI, la información de control fluye en ambas direcciones. Como resultado, se puede llamar a salidas de emisión y de recepción en ambos extremos de un canal.

La unidad de datos que se transmite en un solo flujo entre dos MCA se denomina *segmento de transmisión*. Las salidas de emisión y recepción tienen acceso a cada segmento de transmisión. Pueden modificar su contenido y cambiar su longitud. Sin embargo, una salida de emisión no debe cambiar los 8 primeros bytes de un segmento de transmisión. Estos 8 bytes forman parte de la cabecera del protocolo de canal de IBM MQ. También hay restricciones en la cantidad en que una salida de emisión puede aumentar la longitud de un segmento de transmisión. En concreto, una salida de emisión no puede aumentar su longitud por encima del máximo negociado entre los dos MCA en el momento del inicio del canal.

En un canal de mensajes, si un mensaje es demasiado largo y no se puede enviar en un solo segmento de transmisión, el MCA emisor divide el mensaje y lo envía en más de un segmento de transmisión. Como consecuencia, se llama a una salida de emisión para cada segmento de transmisión que contiene una parte del mensaje y, en el extremo receptor, se llama a una rutina de recepción para cada segmento de transmisión. El MCA receptor vuelve a construir el mensaje a partir de los segmentos de transmisión después de que la salida de recepción los haya procesado.

De forma similar, en un canal MQI, los parámetros de entrada o salida de una llamada MQI se envían en más de un segmento de transmisión si son demasiado largos. Esto puede suceder, por ejemplo, en una llamada MQPUT, MQPUT1 o MQGET si los datos de aplicación son lo suficientemente grandes.

Teniendo esto en cuenta, es más adecuado utilizar salidas de emisión y recepción en casos en que no tengan que comprender la estructura de los datos que manejan y puedan, por tanto, tratar cada segmento de transmisión como un objeto binario.

Una salida de emisión o de recepción puede cerrar un canal.

Los nombres de una salida de emisión y de una de recepción se especifican como parámetros en la definición de canal en cada extremo de un canal. También puede especificar una lista de salidas de emisión para que se ejecuten en sucesión. De forma similar, puede especificar una lista de salidas de recepción.

Para obtener más información acerca de las salidas de recepción y de emisión, consulte la publicación “Seguridad a nivel de enlace mediante salidas de emisión y recepción” en la página 104.

Planificación de la integridad de datos

Planifique cómo preservar la integridad de los datos.

Puede implementar la integridad de datos a nivel de la aplicación o a nivel del enlace.

A nivel de aplicación, también puede utilizar los programas de salida de API si los recursos estándares no satisfacen los requisitos. Puede optar por utilizar Advanced Message Security (AMS) para firmar digitalmente los mensajes para protegerse con el fin de protegerlos frente la modificación no autorizada.

A nivel de enlace, puede optar por utilizar TLS, en cuyo caso debe planificar el uso de certificados digitales. También puede utilizar programas de salida de canal si los recursos estándares no satisfacen los requisitos.

Conceptos relacionados

"Protección de canales con SSL/TLS" en la página 113
El soporte de TLS en IBM MQ utiliza el objeto de información de autenticación del gestor de colas y diversos mandatos MQSC. También debe contemplar el uso de certificados digitales.

"Integridad de los datos en IBM MQ" en la página 23
Puede utilizar un servicio de integridad de datos para detectar si se ha modificado un mensaje.

"Planificación de Advanced Message Security" en la página 105
Advanced Message Security (AMS) es un componente de IBM MQ que proporciona un alto nivel de protección para los datos confidenciales que fluyen a través de la red IBM MQ, aunque no afecta a las aplicaciones finales.

Llamadas de salida de canal y estructuras de datos

Referencia relacionada

Referencia a la salida de la API

Planificación de la auditoría

Decida qué datos necesita auditar, y cómo va a capturar y procesar información de auditoría. Tenga en cuenta cómo comprobar que el sistema está configurado correctamente.

Hay varios aspectos para la supervisión de la actividad. Los aspectos que debe tener en cuenta a menudo los definen los requisitos del auditor, y estas necesidades a menudo se controlan por los estándares normativos como HIPAA (Health Insurance Portability and Accountability Act) o SOX (Sarbanes-Oxley). IBM MQ proporciona características destinadas a ayudar con la conformidad con los estándares.

Considere si sólo está interesado en las excepciones o si está interesado en todo el comportamiento del sistema.

Algunos aspectos de la auditoría también pueden considerarse como la supervisión operativa; una distinción para la auditoría es que a menudo están examinando los datos históricos, no solo examinar las alertas en tiempo real. La supervisión se describe en la sección Supervisión y rendimiento.

¿Qué datos deben auditarse?

Considere qué tipos de datos o actividad es necesario auditar, tal como se describe en las secciones siguientes:

Los cambios realizados en IBM MQ utilizando las interfaces de IBM MQ
Configure IBM MQ para emitir sucesos de instrumentación, específicamente sucesos de mandatos y sucesos de configuración.

Los cambios realizados en IBM MQ fuera de su control
Algunos cambios pueden afectar a cómo se comporta IBM MQ, pero no pueden supervisarse directamente mediante IBM MQ. Algunos ejemplos de esos cambios incluyen cambios en la configuración de los archivos `mqs.ini`, `qm.ini`y `mqclient.ini`, la creación y supresión de gestores de colas, la instalación de los archivos binarios como programas de salida de usuario, y los cambios en los permisos de archivos. Para supervisar estas actividades, debe utilizar herramientas que se ejecutan en el nivel del sistema operativo. Hay diferentes herramientas disponibles y apropiadas para sistemas operativos diferentes. También puede tener registros creados por las herramientas asociadas como *sudo*.

El control operacional de IBM MQ
Puede utilizar las herramientas del sistema operativo para auditar actividades como el inicio y la detención de gestores de colas. En algunos casos, IBM MQ se puede configurar para emitir sucesos de instrumentación.

La actividad de aplicación dentro de IBM MQ
Para auditar las acciones de aplicaciones, por ejemplo la apertura de colas y la transferencia y obtención de mensajes, configure IBM MQ para emitir los sucesos adecuados.

Alertas de intrusos
Para auditar las vulneraciones de la seguridad que se han intentado, configure el sistema para emitir sucesos de autorización. Los sucesos de canal también podrían ser útiles para mostrar actividad, especialmente si un canal finaliza inesperadamente.

Planificación de la captura, la visualización y el archivado de datos de auditoría

Muchos de los elementos necesarios se notifican como mensajes de sucesos de IBM MQ. Debe elegir herramientas que puedan leer y formatear estos mensajes. Si está interesado en el almacenamiento y análisis a largo plazo debe trasladarlos a un mecanismo de almacenamiento auxiliar como una base de datos. Si no procesa estos mensajes, estos permanecen en la cola de sucesos, posiblemente llenando la cola. Puede decidir implementar una herramienta que actúe automáticamente basándose en algunos sucesos; por ejemplo, emitir una alerta cuando se produce un fallo de seguridad.

Verificación de que el sistema está configurado correctamente

Se facilitan un conjunto de pruebas con IBM MQ Explorer. Utilícelas para comprobar si hay problemas en las definiciones de objetos.

Asimismo, compruebe periódicamente que la configuración del sistema es la que espera. Aunque los sucesos de mandatos y configuración pueden notificar cuando algo se modifica, también es útil para volcar la configuración y compararla con un buena copia conocida.

Planificación de seguridad según topología

En esta sección se describe la seguridad en situaciones específicas, en concreto de los canales, los clústeres de gestores de colas, las aplicaciones de publicación/suscripción y multidifusión, y cuando se utiliza un cortafuegos.

Consulte los subtemas siguientes para obtener más información:

Autorización de canal

Al enviar o recibir un mensaje a través de un canal, es necesario proporcionar acceso a diversos recursos de IBM MQ. Los agentes de canal de mensajes (MCA) son fundamentalmente aplicaciones IBM MQ que mueven mensajes entre gestores de colas y como tales requieren acceder a diversos recursos de IBM MQ para funcionar correctamente.

Para recibir mensajes en la hora de transferencia para los MCA, puede utilizar el ID de usuario asociado al MCA, o el ID de usuario asociado al mensaje.

En la hora de conexión puede correlacionar el ID de usuario certificado con un usuario alternativo, utilizando los registros de autenticación de canal **CHLAUTH**.

En IBM MQ, los canales pueden protegerse mediante el soporte TLS.

Los ID de usuario asociados con los canales emisores y receptores, excluido el canal emisor donde no se utiliza el atributo MCAUSER, requieren acceder a los siguientes recursos:

- El ID de usuario asociado a un canal emisor requiere acceso al gestor de colas, la cola de transmisión, la cola de mensajes no entregados y el acceso a los demás recursos requeridos por las salidas de canal.
- El ID de usuario MCAUSER de un canal receptor necesita la autorización *+setall*. La razón es que el canal receptor tiene que crear el MQMD completo, incluidos los campos de contexto, utilizando los datos que ha recibido del canal emisor remoto. Por lo tanto el gestor de colas requiere que el usuario que lleve a cabo esta actividad tenga la autorización *+setall*. Esta autorización *+setall* debe otorgarse al usuario para:
  - Todas las colas en las que el canal receptor coloca válidamente los mensajes.

  - El objeto de gestor de colas. Para obtener más información, consulte Autorizaciones para contexto.
- El ID de usuario MCAUSER de un canal receptor donde el originador ha solicitado un mensaje de informe COA necesita autorización *+passid* en la cola de transmisión que devuelve el mensaje de informe. Sin esta autorización, se anotan mensajes de error AMQ8077.
- Con el ID de usuario asociado al canal receptor, se pueden abrir las colas de destino para poner mensajes en ellas. Esto implica el uso de la MQI (interfaz de cola de mensajes), por lo tanto, es posible que sea necesario realizar comprobaciones de control de acceso adicionales si no está utilizando el Gestor de autorizaciones sobre objetos (OAM) de IBM MQ. Puede especificar si las comprobaciones de autorización se realizan en el ID de usuario asociado al MCA (tal como se describe en este tema) o en el ID de usuario asociado al mensaje desde el campo UserIdentifier de MQMD).

  Para los tipos de canal a los que se aplica, el parámetro **PUTAUT** de la definición de un canal especifica qué ID de usuario se utiliza para estas comprobaciones.

  - De forma predeterminada, el canal utiliza la cuenta de servicio del gestor de colas, que tiene plenos derechos administrativos y no requiere autorizaciones especiales.
  - En el caso e canales de conexión de servidor, las conexiones administrativas se bloquean de forma predeterminada por las reglas CHLAUTH y requieren un suministro explícito.
  - Los canales de tipo receptor, peticionario y receptor en clúster permiten que cualquier gestor de colas adyacente realice la administración local, a menos el administrador tome medidas para restringir este acceso.
  - No es necesario otorgar autoridad *dsp* y *ctrlx* al ID de usuario MCAUSER de un canal receptor.
  - Antes de IBM MQ 8.0.0 Fix Pack 4, si se utiliza un ID de usuario al que le faltan privilegios administrativos de IBM MQ, hay que otorgar la autorización **dsp** y **ctrlx** a dicho ID de usuario para que el canal funcione.

    Desde IBM MQ 8.0.0 Fix Pack 4, no existe ninguna comprobación de autoridad cuando un canal se resincroniza a sí mismo y corrige números de secuencia.

    Sin embargo, la emisión de un mandato RESET CHANNEL sigue necesitando **+dsp** y **+ctrlx** en todos los releases.

    Atención: Cuando se requiere un restablecimiento de canal para la confirmación por lotes de mensajes, IBM MQ intenta consultar el canal, que requiere autorización **+dsp**.
  - El atributo MCAUSER no se utiliza para el tipo de canal SDR.
  - Si utiliza el ID de usuario asociado al mensaje, es probable que el ID de usuario proceda de un sistema remoto. Este ID de usuario del sistema remoto debe ser reconocido por el sistema de destino. Los mandatos siguientes son ejemplos del tipo de mandato que se puede ejecutar para otorgar autoridad a un ID de usuario de un sistema remoto:

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

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

donde *Perfil* es un canal.

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

donde *Perfil* es una cola de mensajes no entregados, si está configurada.

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

donde *Perfil* es una lista de colas autorizadas.

Atención: Tenga cuidado al autorizar un ID de usuario para que coloque mensajes en la cola de mandatos u otras colas del sistema sensibles.

El ID de usuario asociado al MCA depende del tipo del MCA. Hay dos tipos de MCA:

MCA de llamada
Los MCA que inician un canal. Los MCA de llamada se pueden iniciar como procesos individuales, como hebras del iniciador de canal o como hebras de una agrupación de procesos. El ID de usuario utilizado es el ID de usuario asociado al proceso padre (el iniciador de canal) o el ID de usuario asociado con el proceso que inicia al MCA.

MCA de respuesta
Los MCA de respuesta son los MCA que se inician como resultado de una solicitud del MCA de llamada. Los MCA de respuesta se pueden iniciar como procesos individuales, como hebras del escucha o como hebras de una agrupación de procesos. El ID de usuario puede ser cualquiera de los tipos siguientes (en este orden de preferencia):

1. En APPC, el MCA de llamada puede indicar el ID de usuario que se debe utilizar para el MCA de respuesta. Esto se denomina el ID de usuario de red y se aplica solamente a los canales que se inician como procesos individuales. Establezca el ID de usuario de red ce con el parámetro USERID de la definición de canal.
2. Si no se utiliza el parámetro **USERID**, la definición de canal del MCA de respuesta puede especificar el ID de usuario que debe utilizar el MCA. Establezca el ID de usuario mediante el parámetro **MCAUSER** de la definición de canal.
3. Si el ID de usuario no se ha establecido siguiendo ninguno de los (dos) métodos anteriores, se utiliza el ID de usuario del proceso que inicia MCA o el ID de usuario del proceso padre (el escucha).

Conceptos relacionados

"Registros de autenticación de canal" en la página 49
Para ejercer un control más preciso sobre el acceso otorgado a la conexión de sistemas a nivel de canal, puede utilizar registros de autenticación de canal.

Propiedades del registro de autenticación de canal

Protección de las definiciones de iniciador de canal

Sólo los miembros del grupo mqm pueden manipular iniciadores de canal.

Los iniciadores de canal de IBM MQ no son objetos IBM MQ; el OAM no controla el acceso a los mismos. IBM MQ no permite que los usuarios ni las aplicaciones manipulen estos objetos a menos que su ID de usuario sea miembro del grupo mqm. Si tiene una aplicación que emite el mandato PCF **StartChannelInitiator**, el ID de usuario especificado en el descriptor de mensaje del mensaje PCF debe ser miembro del grupo mqm en el gestor de colas de destino.

Un ID de usuario también debe ser miembro del grupo mqm en la máquina de destino para emitir los mandatos MQSC equivalentes mediante el mandato PCF de Escape o utilizando `runmqsc` en modalidad indirecta.

Colas de transmisión

Los gestores de colas transfieren automáticamente los mensajes remotos a una cola de transmisión; para ello no se requiere ninguna autorización especial.

Sin embargo, si tiene que transferir un mensaje directamente a una cola de transmisión, se necesita una autorización especial; consulte Tabla 12 en la página 131.

Salidas de canal

Si los registros de autenticación de canal no resultan adecuados, puede utilizar las salidas de canal para obtener una mayor seguridad. Una salida de seguridad forma una conexión segura entre dos programas de salida de seguridad. Un programa es para el agente de canal de mensajes (MCA) emisor y el otro es para el MCA receptor.

Consulte "Programas de salida de canal" en la página 106 para obtener más información sobre las salidas de canal.

Protección de canales con SSL/TLS

El soporte de TLS en IBM MQ utiliza el objeto de información de autenticación del gestor de colas y diversos mandatos MQSC. También debe contemplar el uso de certificados digitales.

Certificados digitales y depósitos de claves

Se recomienda establecer el atributo de etiqueta de certificado del gestor de colas (**CERTLABL**) en el nombre del certificado personal que se va a utilizar para la mayor parte de los canales, y alterarlo temporalmente en el caso de excepciones, estableciendo la etiqueta de certificado en aquellos canales que requieren certificados diferentes.

Si necesita muchos canales con certificados que difieren del conjunto de certificados predeterminado en el gestor de colas, debe considerar la posibilidad de dividir los canales entre varios gestores de colas o utilizar un proxy MQIPT frente al gestor de colas para presentar un certificado diferente.

Puede utilizar un certificado diferente para cada canal, pero si almacena demasiados certificados en un depósito de claves, el rendimiento puede verse afectado cuando se inician los canales TLS. Intente mantener en un depósito de claves un número de certificados que no sea inferior a 50 aproximadamente y considere 100 como el número máximo, ya que el rendimiento de GSKit disminuye notablemente en los depósitos de claves de gran tamaño.

Si se permiten varios certificados en el mismo gestor de colas se aumentan las posibilidades de que se utilizarán varios certificados de CA en el mismo gestor de colas. De este modo, se aumentan las posibilidades de que existan conflictos en los espacios de nombres distinguidos del asunto para los certificados emitidos por diferentes entidades emisoras de certificados.

Aunque es probable que las entidades emisoras de certificados profesionales tengan mucho cuidado, normalmente las entidades emisoras de certificados internas no tienen convenios de denominación claros y pueden producirse incoherencias no intencionadas entre una entidad emisora de certificados y otra.

Debe comprobar el nombre distinguido del emisor del certificado además del nombre distinguido del asunto. Para ello,utilice un registro SSLPEERMAP de autenticación y establezca los campos **SSLPEER** y **SSLCERTI** de modo que coincidan con el nombre distinguido del asunto y del emisor respectivamente.

Certificados autofirmados y firmados por CA

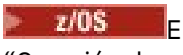
Es importante planificar el uso de certificados digitales, tanto cuando se está desarrollando y probando la aplicación y para su uso en producción. Puede usar certificados firmados por CA o certificados autofirmados, en función del uso de los gestores de colas y las aplicaciones cliente.

Certificados firmados por CA
Para los sistemas de producción, obtenga los certificados de una autoridad certificadora de confianza (CA). Cuando obtiene un certificado de una CA externa, debe pagar por el servicio.

Certificados autofirmados
Mientras desarrolla la aplicación puede utilizar certificados autofirmados o certificados emitidos por una CA local, según la plataforma:

En los sistemas Windows, UNIX y Linux, puede usar certificados autofirmados. Consulte “Creación de un certificado personal autofirmado en UNIX, Linux, and Windows” en la página 298 para obtener instrucciones.

En los sistemas IBM i, puede utilizar los certificados firmados por la CA local. Consulte “Solicitar un certificado de servidor en IBM i” en la página 282 para obtener instrucciones.



En z/OS, puede utilizar certificados autofirmados o firmados por CA local. Consulte “Creación de un certificado personal autofirmado en z/OS” en la página 326 o “Solicitud de un certificado personal en z/OS” en la página 327 para obtener instrucciones.

Los certificados autofirmados no son adecuados para el uso en producción por las siguientes razones:

- Los certificados autofirmados no se pueden revocar, lo que podría permitir a un atacante suplantar una identidad después de que se haya comprometido una clave privada. Las CA pueden revocar un certificado comprometido, lo que impide su posterior uso. Los certificados firmados por una CA son, por lo tanto, más seguros para su uso en un entorno de producción, aunque los certificados autofirmados son más convenientes para un sistema de prueba.
- Los certificados autofirmados nunca caducan. Esto es cómodo y seguro en un entorno de prueba, pero en un entorno de producción pueden producirse infracciones de seguridad. El riesgo se agrava por el hecho de que los certificados autofirmados no se pueden revocar.
- Un certificado autofirmado se utiliza como un certificado personal y como un certificado de CA raíz (o ancla de confianza del certificado). Un usuario con un certificado personal autofirmado podría utilizarlo para firmar otros certificados personales. En general, esto no se cumple en certificados personales emitidos por una CA y representa un riesgo significativo.

CipherSpecs y certificados digitales

Únicamente un subconjunto de las CipherSpecs soportadas puede utilizarse con todos los tipos de certificados digitales soportados. Por consiguiente, es necesario que elija una CipherSpec adecuada para su certificado digital. Del mismo modo, si la política de seguridad de la empresa requiere que se utilice una determinada CipherSpec, debe obtener certificados digitales adecuados.

Para obtener más información sobre la relación entre CipherSpecs y los certificados digitales, consulte “Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45

Políticas de validación de certificados

El estándar IETF RFC 5280 especifica una serie de reglas de validación de certificados que el software de aplicación compatible debe implementar para evitar ataques de suplantación. Un conjunto de reglas de validación de certificados se conoce como una política de validación de certificados. Para obtener más información sobre las políticas de validación de certificados en IBM MQ, consulte “Políticas de validación de certificados en IBM MQ” en la página 44.

Planificación de la comprobación de la revocación de certificados

Si se permiten varios certificados de diferentes entidades emisoras de certificados es muy probable que se requiera una comprobación adicional de la revocación de certificados.

En concreto, si ha configurado explícitamente el uso de un servidor de revocación desde una CA específica, por ejemplo, utilizando un objeto AUTHINFO o una estructura MQAIR (Authentication Information Record), la comprobación de la revocación fallará cuando se presente un certificado de una CA diferente.

Debe evitar esta configuración explícita del servidor de revocación de certificados. En su lugar, debe habilitar la comprobación implícita en la que cada certificado contiene su propia ubicación del servidor de revocación en una extensión del certificado, por ejemplo, un punto de distribución de CRL o AuthorityInfoAccess de OCSP.

Para obtener más información, consulte las secciones OCSPCheckExtensions y CDPCheckExtensions.

Mandatos y atributos para soporte TLS

El protocolo TLS (seguridad de la capa de transporte) proporciona seguridad de canal, con protección contra escuchas y manipulaciones no autorizadas y contra falsas identidades. El soporte de IBM MQ para TLS le permite especificar, en la definición de canal, que un canal determinado utilice seguridad TLS. También puede especificar información detallada sobre el tipo de seguridad que desea, como por ejemplo, el algoritmo de cifrado que desea utilizar.

- Los mandatos MQSC siguientes dan soporte a TLS:

  ALTER AUTHINFO
  Modifica los atributos de un objeto de información de autenticación.

DEFINE AUTHINFO
Crea un objeto de información de autenticación.

DELETE AUTHINFO
Suprime un objeto de información de autenticación.

DISPLAY AUTHINFO
Visualiza los atributos de un objeto de información de autenticación específico.

- Los siguientes parámetros de gestor de colas dan soporte a TLS:

CERTLABL
Define una etiqueta de certificado personal que utilizar.

SSLCRLNL
El atributo SSLCRLNL especifica una lista de nombres de objetos de información de autenticación que se utilizan para proporcionar ubicaciones de revocación de certificados para permitir la comprobación de certificados TLS mejorada.

SSLCRYP
En sistemas Windows y UNIX and Linux, establece el atributo de gestor de colas **SSLCryptoHardware**. Este atributo es el nombre de la serie de parámetros que puede utilizar para configurar el hardware criptográfico que tiene en el sistema.

SSLEV
Determina si se envía un mensaje de suceso TLS cuando un canal que utiliza TLS no puede establecer una conexión TLS correctamente.

SSLFIPS
Especifica si sólo se deben utilizar algoritmos certificados por FIPS si el cifrado se lleva a cabo en IBM MQ, en lugar de en el hardware de cifrado. Si el hardware de cifrado está configurado, se utilizan los módulos de cifrado que proporciona el producto de hardware, que pueden estar certificados por FIPS en un nivel determinado. Depende del producto de hardware que se esté utilizando.

SSLKEYR
En sistemas UNIX, Linux, and Windows, asocia un repositorio de claves a un gestor de colas. La base de datos de claves se guarda en una base de datos de claves *GSKit*. IBM Global Security Kit (GSKit) permite utilizar la seguridad TLS en los sistemas Windows y UNIX and Linux.

SSLRKEYC
El número de bytes que se deben enviar y recibir en una conversación TLS antes de volver a negociar la clave secreta. El número de bytes incluye la información de control que envía el MCA.

- Los siguientes parámetros de canal dan soporte a TLS:

CERTLABL
Define una etiqueta de certificado personal que utilizar.

SSLCAUTH
Define si IBM MQ requiere y valida un certificado del cliente TLS.

SSLCIPH
Especifica la fuerza del cifrado y su función (CipherSpec), por ejemplo, TLS_RSA_WITH_AES_128_CBC_SHA. CipherSpec debe coincidir en ambos extremos del canal.

SSLPEER
Especifica el nombre distinguido (identificador exclusivo) de los asociados permitidos.

En este apartado se describen los mandatos **setmqaut**, **dspmqaut**, **dmpmqaut**, **rcrmqobj**, **rcdmqimg** y **dspmqfls** para dar soporte al objeto de información de autenticación. También describe el mandato **runmqckm** (iKeycmd) para gestionar certificados en sistemas UNIX and Linux y la herramienta **runmqakm** para gestionar certificados en UNIX, Linux, and Windows. Consulte los apartados siguientes:

- setmqaut
- dspmqaut
- dmpmqaut

- rcrmqobj
- rcdmqimg
- dspmqfls
- Gestión de claves y certificados

Para obtener una visión general de la seguridad de canal utilizando TLS, consulte

- “Protocolos de seguridad TLS en IBM MQ” en la página 24

Para conocer detalles de los mandatos MQSC asociados a TLS, consulte

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

Para conocer detalles de los mandatos PCF asociados a TLS, consulte

- Cambiar, copiar y crear un objeto de información de autenticación
- Suprimir objeto de información de autenticación
- Consultar objeto de información de autenticación

z/OS *Canal de conexión del servidor de IBM MQ for z/OS*

El canal IBM MQ for z/OS SVRCONN no es seguro si no implementa la autenticación de canal, o si no añade una salida de seguridad utilizando TLS. Los canales SVRCONN no tienen definida una salida de seguridad de forma predeterminada.

Aspectos de seguridad

Los canales SVRCONN no son seguros tal como están definidos inicialmente, por ejemplo SYSTEM.DEF.SVRCONN. Para proteger un canal SVRCONN debe establecer la autenticación de canal mediante el mandato SET CHLAUTH, o instalar una salida de seguridad e implementando TLS.

Debe utilizar una salida de seguridad de ejemplo públicamente disponible, escribir una salida de seguridad o adquirir una salida de seguridad.

Hay diversos ejemplos disponibles que puede utilizar como punto de partida para escribir su propia salida de seguridad de canal SVRCONN.

En IBM MQ for z/OS, el miembro CSQ4BCX3 de la biblioteca hlq.SCSQC37S es una salida de seguridad de ejemplo escrita en lenguaje C. CSQ4BCX3 también viene precompilado en la biblioteca hlq.SCSQAUTH.

Puede implementar la salida de ejemplo CSQ4BCX3 copiando el miembro compilado hlq.SCSQAUTH(CSQ4BCX3) en una biblioteca de carga ubicada en la sentencia CSQXLIB DD del procedimiento CHIN. Tenga en cuenta que CHIN requiere que la biblioteca de carga esté definida como "Controlada por programa".

Altere el canal SVRCONN para definir CSQ4BCX3 como salida de seguridad.

V 9.1.4 Cuando un cliente se conecte utilizando ese canal SVRCONN, CSQ4BCX3 se autenticará utilizando el par **RemoteUserIdentifier** y **RemotePassword** de MQCD o, a partir de IBM MQ 9.1.4, el par **CSPUserIdPtr** y **CSPPasswordPtr** de MQCSP. Si la autenticación es satisfactoria, copiará **RemoteUserIdentifier** en **MCAUserIdentifier**, cambiando el contexto de identidad de la hebra.

Para Long Term Support y Continuous Delivery antes de IBM MQ 9.1.4, cuando un cliente se conecta utilizando ese canal SVRCONN, CSQ4BCX3 se autenticará utilizando el par **RemoteUserIdentifier** y **RemotePassword** de MQCD. Si la autenticación es satisfactoria, copiará **RemoteUserIdentifier** en **MCAUserIdentifier**, cambiando el contexto de identidad de la hebra.

Si está escribiendo un cliente de IBM MQ Java , puede utilizar ventanas emergentes para consultar al usuario y establecer MQEnvironment.userID y MQEnvironment.password. Estos valores se pasarán cuando se realice la conexión.

Ahora que ya tiene una salida de seguridad funcional, tiene el problema adicional de que el ID de usuario y la contraseña se transmiten en texto sin formato por la red cuando se realiza la conexión, igual que el contenido de los subsiguientes mensajes de IBM MQ. Puede utilizar TLS para cifrar esta información de conexión inicial, así como el contenido de cualquier mensaje de IBM MQ .

Ejemplo

Para proteger el canal IBM MQ Explorer SVRCONN SYSTEM.ADMIN.SVRCONN complete los pasos siguientes:

1. Copie hlq.SCSQAUTH(CSQ4BCX3) en una biblioteca de carga ubicada en la sentencia CSQXLIB DD del procedimiento CHINIT.
2. Compruebe que la biblioteca de carga esté controlada por programa.
3. Altere SYSTEM ADMIN.SVRCONN de modo que utilice la salida de seguridad CSQ4BCX3.
4. En IBM MQ Explorer, pulse con el botón derecho del ratón en el nombre del gestor de colas z/OS, seleccione **Connection Details** > **Properties** > **Userid** y especifique el ID de usuario de z/OS.
5. Para conectarse al gestor de colas z/OS, escriba una contraseña.

Información adicional

Para que la salida CSQ4BCX3 se ejecute en un entorno controlado por programa, todo aquello que se haya cargado en el espacio de direcciones de CHIN se debe haber cargado desde una biblioteca controlada por programa, por ejemplo, todas las bibliotecas de STEPLIB y cualquier biblioteca mencionada en CSQXLIB DD. Para definir una biblioteca de carga como controlada por programa emita mandatos de RACF. En el ejemplo siguiente el nombre de la biblioteca de carga es MY.TEST.LOADLIB.

```
RALTER PROGRAM * ADDMEM('MY.TEST.LOADLIB'//NOPADCHK)
SETROPTS WHEN(PROGRAM)REFRESH
```

Para modificar el canal SVRCONN para implementar CSQ4BCX3, emita el siguiente mandato IBM MQ :

```
ALTER CHANNEL(SYSTEM ADMIN.SVRCONN) CHLTYPE(SVRCONN) SCYEXIT(CSQ4BCX3)
```

En el ejemplo anterior, el nombre de canal SVRCONN utilizado es SYSTEM ADMIN.SVRCONN.

Consulte “Programas de salida de canal” en la página 106 para obtener más información sobre las salidas de canal.

Tareas relacionadas

Escritura de programas de salida de canal en z/OS

Servicios de seguridad SNA LU 6.2

SNA LU 6.2 ofrece cifrado a nivel de sesión, autenticación a nivel de sesión y autenticación a nivel de conversación.

Nota: En esta colección de temas se presupone que tiene conocimientos básicos sobre la Arquitectura de redes de sistemas (SNA). La otra documentación a la que se hace referencia en esta sección contiene una breve introducción a los conceptos y terminología relevantes. Si necesita una introducción técnica más completa a SNA, consulte el manual *Systems Network Architecture Technical Overview*, GC30-3073.

SNA LU 6.2 proporciona tres servicios de seguridad:

- Cifrado a nivel de sesión
- Autenticación a nivel de sesión
- Autenticación a nivel de conversación

Para el cifrado a nivel de sesión y la autenticación a nivel de sesión, SNA utiliza el algoritmo *Estándar de cifrado de datos (DES, Data Encryption Standard)*. El algoritmo DES es un algoritmo de cifrado de bloques que utiliza una clave simétrica para cifrar y descifrar datos. Tanto el bloque como la clave tienen una longitud de 8 bytes.

Cifrado a nivel de sesión
El *cifrado a nivel de sesión* cifra y descifra datos de sesión mediante el algoritmo DES. Por lo tanto, se puede utilizar para proporcionar un servicio de confidencial de enlace en canales SNA LU 6.2.

Las unidades lógicas (LU) pueden ofrecer cifrado de datos obligatorio (o necesario), cifrado de datos selectivo o ningún cifrado de datos.

En una *sesión de cifrado obligatorio*, una LU cifra todas las unidades de solicitud de datos de salida y descifra todas las unidades de solicitud de datos de entrada.

En una *sesión de cifrado selectivo*, una LU cifra sólo las unidades de solicitud de datos especificadas por el programa de transacción (TP) emisor. La LU emisora señala que los datos están cifrados estableciendo un indicador en la cabecera de la solicitud. Comprobando este indicador, la LU receptora puede indicar qué unidades de solicitud hay que descifrar antes de pasarlas al TP receptor.

En una red SNA, los MCA de IBM MQ son programas de transacciones. Los MCA no solicitan cifrado para ninguno de los datos que envían. Por lo tanto, el cifrado de datos selectivo no constituye una opción; sólo el cifrado de datos obligatorio o ningún cifrado de datos son opciones posibles en una sesión.

Para obtener información sobre cómo implementar el cifrado de datos obligatorio, consulte la documentación correspondiente a su subsistema SNA. Consulte la misma documentación para obtener información sobre formas más potentes de cifrado que quizás pueda utilizar en su plataforma, como el cifrado Triple DES de 24 bytes en z/OS.

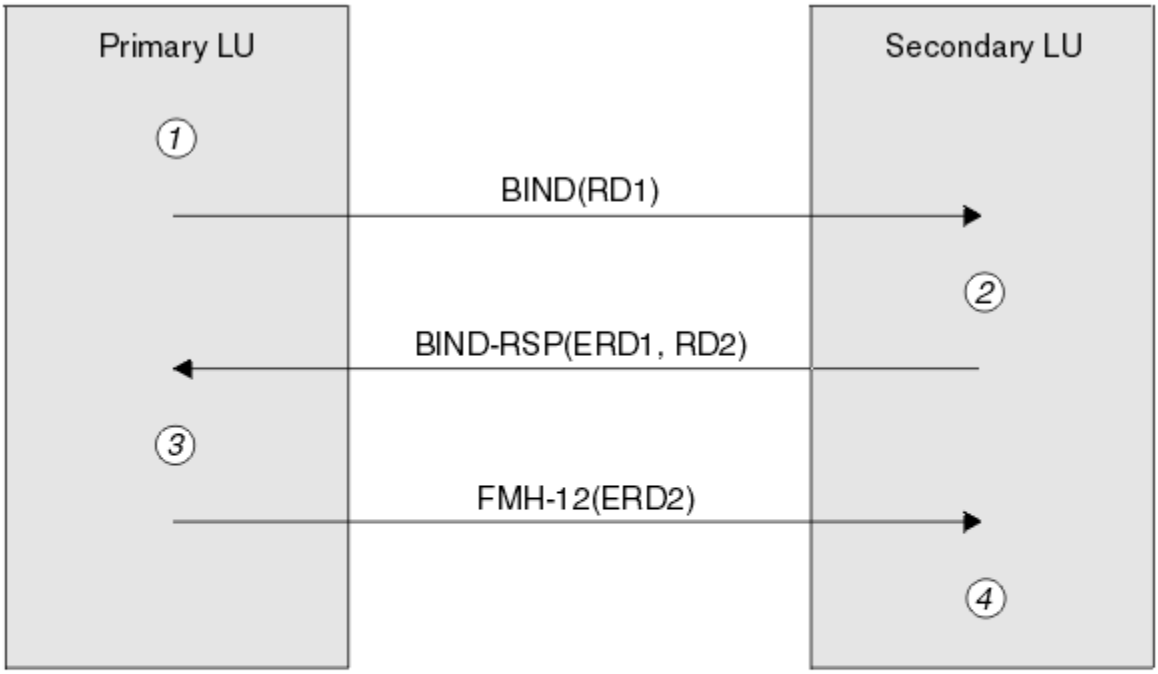
Para obtener información más general sobre el cifrado a nivel de sesión, consulte el manual *Systems Network Architecture LU 6.2 Reference: Peer Protocols* , SC31-6808.

Autenticación a nivel de sesión
La *autenticación a nivel de sesión* es un protocolo de seguridad a nivel de sesión que permite que dos LU se identifiquen entre sí mientras están activando una sesión. También se denomina *verificación LU-LU*.

Puesto que una LU constituye realmente la "pasarela" a un sistema desde la red, es posible que considere que este nivel de autenticación es suficiente en determinadas circunstancias. Por ejemplo, si su gestor de colas tiene que intercambiar mensajes con un gestor de colas remoto que se está ejecutando en un entorno controlado y fiable, es posible que esté preparado para confiar en las identidades de los demás componentes del sistema remoto una vez autenticada la LU.

Cada LU consigue la autenticación a nivel de sesión verificando la contraseña de su asociado. La contraseña se denomina *contraseña LU-LU* porque se establece una contraseña entre cada par de LU. La forma en que se establece una contraseña LU-LU depende de la implementación y queda fuera del ámbito de SNA.

La Figura 12 en la página 119 ilustra los flujos correspondientes a la autenticación a nivel de sesión.



Legend:

BIND	=	BIND request unit
BIND-RSP	=	BIND response unit
ERD	=	Encrypted random data
FMH-12	=	Function Management Header 12
RD	=	Random data

Figura 12. Flujos correspondientes a la autenticación a nivel de sesión

El protocolo correspondiente a la autenticación a nivel de sesión es el siguiente. Los números del procedimiento corresponden a los números de la Figura 12 en la página 119.

1. La LU principal genera un valor de datos aleatorios (RD1) y lo envía a la LU secundaria en la solicitud BIND.
2. Cuando la LU secundaria recibe la solicitud LU con los datos aleatorios, cifra los datos utilizando el algoritmo DES con su copia de la contraseña LU-LU como clave. Luego la LU secundaria genera un segundo valor de datos aleatorios (RD2) y lo envía, con los datos cifrados (ERD1), a la LU principal en la respuesta BIND.
3. Cuando la LU principal recibe la respuesta BIND, calcula su propia versión de los datos cifrados a partir de los datos aleatorios que ha generado originalmente. Para ello utiliza el algoritmo DES con su copia de la contraseña LU-LU como clave. Luego compara su versión con los datos cifrados recibidos en la respuesta BIND. Si los dos valores coinciden, la LU principal sabe que la LU secundaria tiene la misma contraseña que ella y la LU secundaria se autentica. Si los dos valores no coinciden, la LU principal finaliza la sesión.

   Luego la LU principal cifra los datos aleatorios que ha recibido en la respuesta BIND y envía los datos cifrados (ERD2) a la LU secundaria en una Cabecera de gestión de funciones 12 (FMH-12).
4. Cuando la LU secundaria recibe la FMH-12, calcula su propia versión de los datos cifrados a partir de los datos aleatorios que ha generado. Luego compara su versión con los datos cifrados que ha recibido en la FMH-12. Si los dos valores coinciden, la LU principal se autentica. Si los dos valores no coinciden, la LU secundaria finaliza la sesión.

En una versión mejorada del protocolo, que proporciona una mejor protección contra ataques de tipo "man in the middle" (hombre en medio), la LU secundaria calcula un Código de autenticación de mensaje

(MAC) de DES a partir de RD1, RD2 y el nombre completo de la LU secundaria, utilizando su copia de la contraseña LU-LU como clave. La LU secundaria envía el MAC a la LU principal en la respuesta BIND en lugar de ERD1.

La LU principal autentica la LU secundaria, calculando su propia versión del MAC, el cual compara con el MAC recibido en la respuesta BIND. Luego la LU principal calcula un segundo MAC a partir de RD1 y RD2 y envía el MAC a la LU secundaria en la FMH-12 en lugar de ERD2.

La LU secundaria autentica la LU principal calculando su propia versión del segundo MAC, el cual compara con el MAC recibido en la FMH-12.

Para obtener información sobre cómo configurar la autenticación a nivel de sesión, consulte la documentación de su subsistema SNA. Para obtener información más general sobre la autenticación a nivel de sesión, consulte el manual *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808.

Autenticación a nivel de conversación
Cuando un TP local intenta asignar una conversación con un TP asociado, la LU local envía una solicitud de adjuntar a la LU asociada, solicitándole que adjunte el TP asociado. Bajo determinadas circunstancias, la solicitud de adjuntar puede contener información de seguridad, la cual puede utilizar la LU asociada para autenticar el TP local. Esto se denomina *autenticación a nivel de conversación* o *verificación de usuario final*.

Los temas siguientes describen el modo en que IBM MQ proporciona soporte para la autenticación a nivel de conversación.

Para obtener más información acerca de la autenticación a nivel de conversación, consulte el manual *Systems Network Architecture LU 6.2 Reference: Peer Protocols*, SC31-6808. Para obtener información específica de z/OS, consulte *z/OS MVS Planning: APPC/MVS Management*, SA22-7599.

Para obtener más información sobre CPI-C, consulte el manual *Common Programming Interface Communications CPI-C Specification*, SC31-6180. Para obtener más información sobre APPC/MVS TP Conversation Callable Services, consulte el manual *z/OS MVS Programming: Writing Transaction Programs for APPC/MVS*, SA22-7621.

Windows IBM i UNIX *Soporte para la autenticación del nivel de conversación en IBM i, UNIX y Windows*
Utilice este tema para obtener una visión general de cómo funciona la autenticación a nivel de conversación en IBM i, UNIX y Windows.

El soporte para la autenticación del nivel de conversación en IBM i, UNIX y Windows se muestra en Figura 13 en la página 121. Los números del diagrama corresponden a los números de la siguiente descripción.

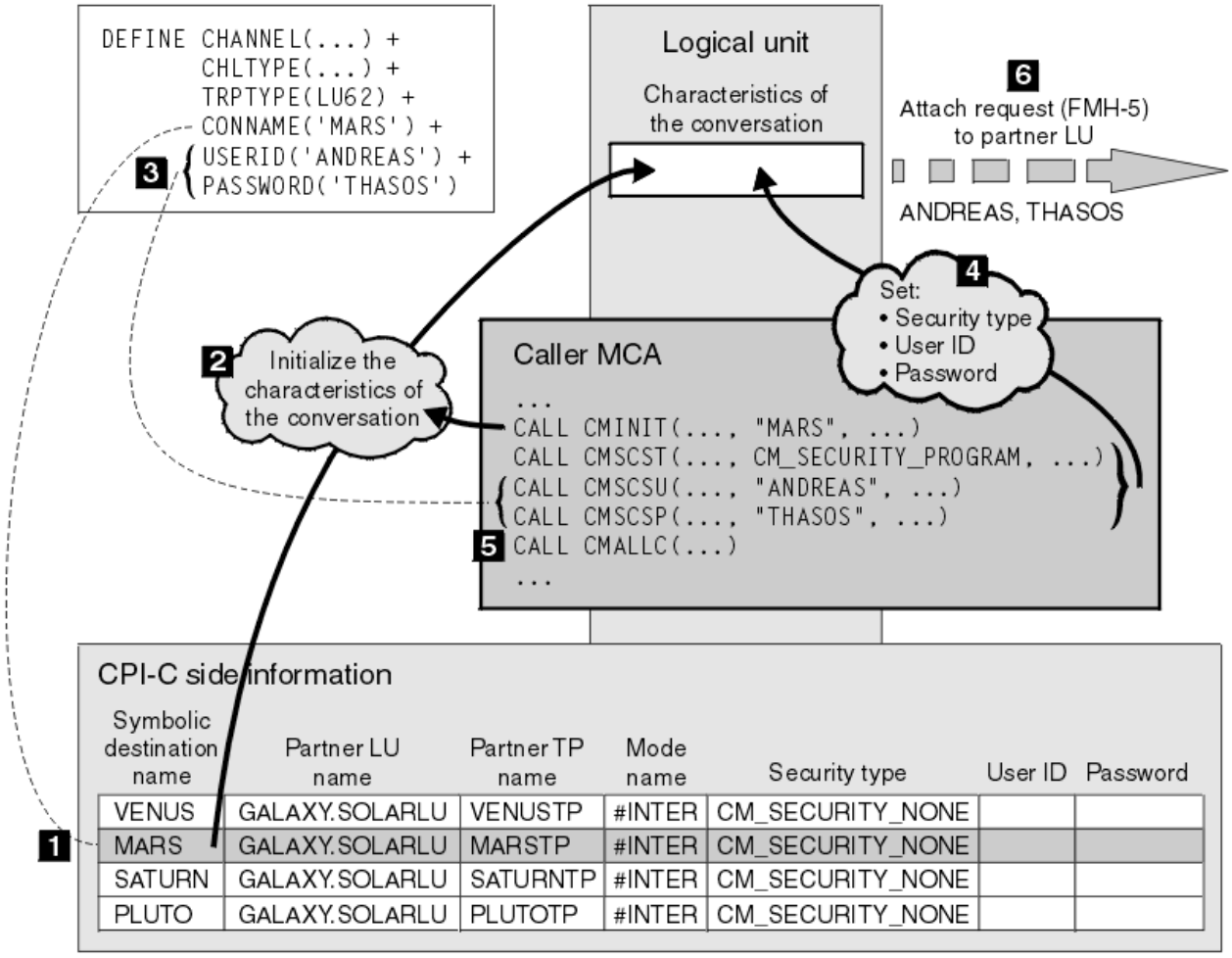


Figura 13. Soporte de IBM MQ para autenticación a nivel de conversación

En IBM i, UNIX y Windows, un MCA utiliza llamadas CPI-C (Common Programming Interface Communications) para comunicarse con un MCA asociado a través de una red SNA. En la definición de canal del extremo de llamada de un canal, el valor del parámetro CONNAME es un nombre de destino simbólico que identifica la entrada de información complementaria de CPI-C (1). Esta entrada especifica:

- El nombre de la LU asociada
- El nombre del TP asociado, que es un MCA de respuesta
- El nombre de la modalidad que se va a utilizar para la conversación

Una entrada de información complementaria también puede especificar la siguiente información de seguridad:

- Un tipo de seguridad.

  Los tipos de seguridad que se suelen implementar son CM_SECURITY_NONE, CM_SECURITY_PROGRAM y CM_SECURITY_SAME, pero hay otros definidos en la especificación CPI-C.
- Un ID de usuario.
- Una contraseña.

Un MCA de llamada se prepara para asignar una conversación con un MCA de respuesta, emitiendo la llamada CPI-C CMINIT, utilizando el valor de CONNAME como uno de los parámetros de la llamada. La llamada CMINIT identifica, como ayuda para la LU local, la entrada de información complementaria que el MCA tiene intención de utilizar para la conversación. La LU local utiliza los valores de esta entrada para inicializar las características de la conversación (2).

Luego el MCA de llamada comprueba los valores de los parámetros USERID y PASSWORD de la definición de canal (3). Si USERID está establecido, el MCA de llamada emite las siguientes llamadas CPI-C (4):

- CMSCST, para establecer el tipo de seguridad correspondiente a la conversación en CM_SECURITY_PROGRAM.
- CMSCSU, para establecer el ID de usuario correspondiente a la conversación en el valor de USERID.
- CMSCSP, para establecer la contraseña correspondiente a la conversación en el valor de PASSWORD. No se llama a CMSCSP a no ser que PASSWORD esté establecido.

El tipo de seguridad, ID de usuario y contraseña establecidos por estas llamadas prevalecen sobre los valores adquiridos previamente de la entrada de información complementaria.

Luego el MCA de llamada emite la llamada CPI-C CMALLC para asignar la conversación (5). En respuesta a esta llamada, la LU local envía una solicitud de adjuntar (Cabecera de gestión de funciones 5 o FMH-5) a la LU asociada (6).

Si la LU asociada acepta un ID de usuario y una contraseña, los valores de USERID y PASSWORD se incluyen en la solicitud de adjuntar. Si la LU asociada no acepta un ID de usuario y contraseña, los valores no se incluyen en la solicitud de adjuntar. La LU local descubre si la LU asociada va a aceptar un ID de usuario y contraseña como parte de un intercambio de información cuando las LU se vinculan para formar una sesión.

En una versión posterior de la solicitud de adjuntar, un sustituto de contraseña puede fluir entre las LU en lugar de una contraseña clara. Un sustituto de contraseña es un Código de autenticación de mensaje (MAC) de DES, o un resumen de mensaje SHA-1, formado a partir de la contraseña. Los sustitutos de contraseña sólo se pueden utilizar si ambas LU les dan soporte.

Cuando la LU asociada recibe una solicitud de adjuntar de entrada que contiene un ID de usuario y una contraseña, puede utilizar el ID de usuario y contraseña con finalidades de identificación y autenticación. Al hacer referencia a listas de control de accesos, la LU asociada también puede determinar si el ID de usuario tiene la autorización para asignar una conversación y adjuntar el MCA de respuesta.

Además, el MCA de respuesta se puede ejecutar bajo el ID de usuario que se incluye en la solicitud de adjuntar.En este caso, el ID de usuario se convierte en el ID de usuario predeterminado para el MCA de respuesta y se utiliza para comprobaciones de autorización cuando el MCA intenta conectar al gestor de colas. También se puede utilizar para siguientes comprobaciones de autorización cuando el MCA intenta acceder a los recursos del gestor de colas.

El modo en que se pueden utilizar un ID de usuario y una contraseña en la solicitud de adjuntar para identificación, autenticación y control de accesos depende de la implementación. Para obtener información específica de su subsistema SNA, consulte la documentación apropiada.

Si USERID no está establecido, el MCA de llamada no llama a CMSCST, CMSCSU ni CMSCSP. En este caso, la información de seguridad que fluye en una solicitud de adjuntar se determina únicamente por lo que se especifica en la entrada de información complementaria y por lo que la LU asociada aceptará.

Autenticación a nivel de conversación e IBM MQ for z/OS
Lea este tema para obtener una visión general de cómo funciona la autenticación a nivel de conversación en z/OS.

En IBM MQ for z/OS, los MCA no utilizan CPI-C. En su lugar, utilizan APPC/MVS TP Conversation Callable Services, una implementación de Advanced Program-to-Program Communication (APPC), que tiene algunas funciones CPI-C. Cuando un MCA de llamada asigna una conversación, se especifica el tipo de seguridad SAME en la llamada. Por lo tanto, puesto que una LU APPC/MVS soporta la verificación permanente únicamente para conversaciones de entrada, no para conversaciones de salida, hay dos posibilidades:

- Si la LU asociada confía en la LU APPC/MVS y aceptará un ID de usuario ya verificado, la LU APPC/MVS envía una solicitud de adjuntar que contiene:
  - El ID de usuario del espacio de direcciones del iniciador de canal
  - Un nombre de perfil de seguridad, el cual, si se utiliza RACF, es el nombre del grupo de conexión actual del ID de usuario de espacio de direcciones del iniciador de canal.

  - Un indicador ya verificado
- Si la LU asociada no confía en la LU APPC/MVS y no aceptará un ID de usuario ya verificado, la LU APPC/MVS envía una solicitud de adjuntar que no contiene información de seguridad.

En IBM MQ for z/OS, los parámetros USERID y PASSWORD en el mandato DEFINE CHANNEL no pueden utilizarse para un canal de mensajes y solamente son válidos en el extremo de conexión de cliente de un canal MQI. Por lo tanto, una solicitud de adjuntar procedente de una LU APPC/MVS nunca contiene valores especificados por estos parámetros.

Seguridad para clústeres de gestores de colas

Aunque puede ser conveniente utilizar clústeres de gestores de colas, debe prestar especial atención a su seguridad.

Un *clúster de gestores de colas* es una red de gestores de colas asociados lógicamente de algún modo. Un gestor de colas que es miembro de un clúster se denomina un *gestor de colas de clúster*.

Una cola que pertenece a un gestor de colas de clúster se puede dar a conocer a otros gestores de colas del clúster. Dicha cola se denomina *cola de clúster*. Cualquier gestor de colas de un clúster puede enviar mensajes a colas de clúster sin necesidad de lo siguiente:

- Una definición de cola remota explícita para cada cola de clúster
- Canales definidos explícitamente a cada gestor de colas remoto y desde cada uno de ellos
- Una cola de transmisión individual para cada canal de salida

Puede crear un clúster en el que dos o varios gestores de colas sean clones. Esto significa que tienen instancias de las mismas colas locales, incluida cualquier cola local declarada como cola de clúster y que puede dar soporte a instancias de las mismas aplicaciones de servidor.

Cuando una aplicación conectada a un gestor de colas de clúster envía un mensaje a una cola de clúster que posee una instancia en cada uno de los gestores de colas clonados, IBM MQ decide a qué gestor de colas lo envía. Cuando muchas aplicaciones envían mensajes a una cola de clúster, IBM MQ equilibra la carga de trabajo entre todos los gestores de colas que poseen una instancia de la cola. Si uno de los sistemas que alberga un gestor de colas clonado sufre una anomalía, IBM MQ continúa equilibrando la carga de trabajo entre los gestores de colas restantes hasta que el sistema anómalo se reinicia.

Si va a utilizar clústeres de gestores de colas, debe tener en cuenta las siguientes cuestiones de seguridad:

- Si permite que solamente los gestores de colas seleccionados envíen mensajes al gestor de colas
- Si permite que solamente los usuarios seleccionados de un gestor de colas remoto envíen mensajes a una cola del gestor de colas
- Si permite que las aplicaciones conectadas al gestor de colas envíen mensajes solamente a las colas remotas seleccionadas

Estas consideraciones son relevantes incluso si no utiliza clústeres, pero resultan más importantes si los está utilizando.

Si una aplicación puede enviar mensajes a una cola de clúster, podrá enviar mensajes a cualquier otra cola de clúster sin necesitar definiciones de colas remotas, colas de transmisión ni canales adicionales. Por lo tanto, resulta más importante considerar si debe limitar el acceso a las colas de clúster en su gestor de colas y limitar las colas de clúster a aquéllas a las que sus aplicaciones pueden enviar mensajes.

Hay algunas consideraciones de seguridad adicionales que resultan relevantes solamente si está utilizando clústeres de gestores de colas:

- Si permite que solamente los gestores de colas seleccionados se unan a un clúster
- Forzar que los gestores de colas no deseados abandonen un clúster

Para obtener más información sobre todas estas consideraciones, consulte Mantenimiento de la seguridad de los clústeres. z/OS Si desea ver consideraciones específicas de IBM MQ for z/OS, consulte “Seguridad de los clústeres de gestores de colas en z/OS” en la página 267.

Tareas relacionadas

“Cómo impedir que los gestores de colas reciban mensajes” en la página 469
Puede impedir que un gestor de colas reciba mensajes si no está autorizado para recibirlos utilizando programas de salida.

Seguridad para Publicación/Suscripción de IBM MQ

Existen consideraciones de seguridad adicionales si está utilizando Publicación/Suscripción de IBM MQ.

En un sistema de publicación/suscripción, hay dos tipos de aplicaciones: el publicador y el suscriptor. Los *publicadores* proporcionan información en forma de mensajes IBM MQ. Cuando un publicador publica un mensaje, especifica un *tema*, que identifica el tema de la información que contiene el mensaje.

Los *suscriptores* son los que consumen la información publicada. Un suscriptor especifica los temas que le interesan suscribiéndose a ellos.

El *gestor de colas* es una aplicación que se suministra con Publicación/Suscripción de IBM MQ. Éste recibe los mensajes que han publicado los publicadores y las peticiones de suscripción de los suscriptores y dirige los mensajes publicados a los suscriptores. A un suscriptor se le envían solamente los mensajes de los temas a los que se ha suscrito.

Para obtener más información, consulte Seguridad de Publicación/suscripción.

Seguridad de multidifusión

Utilice esta información para comprender por qué pueden ser necesarios los procesos de seguridad con IBM MQ Multicast.

IBM MQ Multicast no tiene seguridad incorporada. Las comprobaciones de seguridad se manejan en el gestor de colas en tiempo de MQOPEN y el valor del campo MQMD lo maneja el cliente. Es posible que algunas aplicaciones de la red no sean aplicaciones de IBM MQ (por ejemplo, las aplicaciones LLM, consulte Interoperatividad de multidifusión con mensajes de baja latencia de IBM MQ para obtener más información), tal vez tenga que implementar sus propios procedimientos de seguridad porque las aplicaciones receptoras no pueden estar seguras de la validez de los campos de contexto.

Hay tres procesos de seguridad que se deben tener en cuenta:

Control de accesos

El control de acceso en IBM MQ está basado en los ID de usuario. Para obtener más información sobre este asunto, consulte “Control de accesos para clientes” en la página 98.

Seguridad de red

Una red aislada podría ser una opción de seguridad viable para evitar mensajes falsos. Es posible que una aplicación en la dirección del grupo de multidifusión publique mensajes malintencionados utilizando las funciones de comunicación nativas, que son imposibles de distinguir de los mensajes MQ porque vienen de una aplicación en la misma dirección del grupo de multidifusión.

También es posible que un cliente en la dirección del grupo de multidifusión reciba mensajes que estaban previstos para otros clientes en la misma dirección del grupo de multidifusión.

Aislar la red de multidifusión asegura que sólo los clientes y aplicaciones válidos tienen acceso. Esta precaución de seguridad puede impedir que entren mensajes malintencionados y que salga información confidencial.

Para obtener información sobre las direcciones de red de grupo de multidifusión, consulte: Establecer la red adecuada para el tráfico de multidifusión

Firmas digitales

Una firma digital se crea cifrando una representación de un mensaje. El cifrado utiliza la clave privada del que firma y, por motivos prácticos, suele operar en un resumen del mensaje en lugar de hacerlo en el mensaje propiamente dicho. La firma digital de un mensaje antes de MQPUT es una buena precaución de seguridad, pero este proceso puede tener un efecto perjudicial sobre el rendimiento si hay un gran volumen de mensajes.

Las firmas digitales varían con los datos que se firman. Si la misma entidad firma digitalmente dos mensajes diferentes, las dos firmas serán diferentes pero ambas pueden verificarse con la misma clave pública, es decir, la clave pública de la entidad que ha firmado los mensajes.

Como se ha mencionado anteriormente en esta sección, puede ser posible que una aplicación de la dirección del grupo de multidifusión publique mensajes malintencionados utilizando las funciones de comunicación nativas, que son imposibles de distinguir de los mensajes MQ. Las firmas digitales proporcionan una prueba de origen y solamente el emisor conoce la clave privada, que proporciona una prueba clara de que el remitente es el originador del mensaje.

Para obtener más información sobre este asunto, consulte "Conceptos de cifrado" en la página 7.

Cortafuegos e Internet Pass-thru

Normalmente, se utiliza un cortafuegos para impedir el acceso a direcciones IP hostiles; por ejemplo, en un ataque de denegación de servicio. Sin embargo, es posible que necesite bloquear temporalmente direcciones IP dentro de IBM MQ, quizás mientras espera a que un administrador de seguridad actualice las reglas del cortafuegos.

Para bloquear una o más direcciones IP, cree un registro de autenticación de canal de tipo BLOCKADDR o ADDRESSMAP. Para obtener más información, consulte "Bloquear direcciones IP específicas" en la página 390.

seguridad para IBM MQ Internet Pass-Thru

IBM MQ Internet Pass-Thru puede simplificar la comunicación a través de un cortafuegos, pero esto tiene implicaciones de seguridad.

IBM MQ Internet Pass-Thru (MQIPT) es un componente opcional de IBM MQ que se puede utilizar para implementar soluciones de mensajería entre sitios remotos a través de Internet.

MQIPT permite que dos gestores de colas intercambien mensajes, o que una aplicación cliente IBM MQ se conecte a un gestor de colas, a través de Internet sin necesidad de tener una conexión TCP/IP directa. Resulta útil si un cortafuegos prohíbe la conexión TCP/IP directa entre dos sistemas. Facilita el flujo de entrada y salida a través de un cortafuegos del protocolo de canal de IBM MQ y lo hace más manejable ya que canaliza los flujos en túneles mediante HTTP o actuando como servidor proxy. Mediante TLS (seguridad de la capa de transporte), se puede utilizar también para cifrar y descifrar los mensajes que se envían a través de Internet.

Cuando el sistema IBM MQ se comunique con MQIPT, a menos que se utilice SSL en MQIPT, asegúrese de que la CipherSpec que utiliza IBM MQ coincida con la CipherSuite que utiliza MQIPT:

- Cuando MQIPT actúa como el servidor TLS y IBM MQ se conecta como el cliente TLS, la CipherSpec que utiliza IBM MQ debe corresponder a una CipherSuite que esté habilitada en el conjunto de claves MQIPT relevante.
- Cuando MQIPT actúa como el cliente TLS y se conecta a un servidor TLS de IBM MQ, a CipherSuite de MQIPT debe coincidir con la CipherSpec definida en el canal IBM MQ receptor.

Si migra de MQIPT al soporte TLS de IBM MQ integrado, transfiera los certificados digitales desde el conjunto de claves MQIPT utilizando **mqiptKeyman** o **mqiptKeycmd**.

Para obtener más información, consulte IBM MQ Internet Pass-Thru.

z/OS Lista de comprobación para la implementación de la seguridad de IBM MQ for z/OS

Este tema ofrece un procedimiento paso a paso que puede utilizar para planificar y definir la implementación de seguridad para cada uno de sus gestores de colas de IBM MQ.

RACF proporciona definiciones para las clases de seguridad de IBM MQ en la tabla de descriptores de clase (CDT). A medida que trabaja con la lista de comprobación, puede determinar cuáles de estas clases requiere su configuración. Debe asegurarse de que estén activadas como se describe en "Clases de seguridad de RACF" en la página 186.

Consulte otras secciones para obtener más información, en particular "Perfiles utilizados para controlar el acceso a los recursos de IBM MQ" en la página 196.

Si necesita comprobación de seguridad, siga esta lista de comprobación para implementarla:

1. Active las clases RACF MQADMIN (perfiles en mayúsculas) o MXADMIN (perfiles que combinan mayúsculas y minúsculas).
   - ¿Desea seguridad a nivel de grupo de compartición de colas, a nivel de gestor de colas o una combinación de ambos?

     Consulte "Perfiles para controlar la seguridad a nivel de grupo de compartición de colas o de gestor de colas" en la página 191.
2. ¿Necesita seguridad de conexión?
   - **Sí**: Active la clase MQCONN. Defina perfiles de conexión adecuados a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQCONN. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.

     Nota: Sólo los usuarios de la solicitud API `MQCONN` o los ID de usuario del espacio de direcciones de CICS o IMS necesitan tener acceso al perfil de conexión correspondiente.
   - **No**: Defina un perfil hlq.NO.CONNECT.CHECKS a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQADMIN o MXADMIN.
3. ¿Necesita comprobación de seguridad en los mandatos?
   - **Sí**: Active la clase MQCMDS. Defina perfiles de mandato adecuados a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQCMDS. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.

     Si está utilizando un grupo de compartición de colas, es posible que necesite incluir los ID de usuario que utilizan el propio gestor de colas y el iniciador de canal. Consulte "Configuración de la seguridad de recursos de IBM MQ for z/OS" en la página 258.
   - **No**: Defina un perfil hlq.NO.CMD.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
4. ¿Necesita seguridad en los recursos que se utilizan en mandatos?
   - **Sí**: Asegúrese de que la clase MQADMIN o MXADMIN esté activa. Defina perfiles adecuados para proteger los recursos en los mandatos a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQADMIN o MXADMIN. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles. Establezca el parámetro CMDUSER de CSQ6SYSP en el ID de usuario predeterminado que se utilizará para las comprobaciones de seguridad de mandatos.

     Si está utilizando un grupo de compartición de colas, es posible que necesite incluir los ID de usuario que utilizan el propio gestor de colas y el iniciador de canal. Consulte "Configuración de la seguridad de recursos de IBM MQ for z/OS" en la página 258.
   - **No**: Defina un perfil hlq.NO.CMD.RESC.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
5. ¿Necesita seguridad de colas?

- **Sí**: Active la clase MQQUEUE o MXQUEUE. Defina perfiles de cola adecuados para el gestor de colas o el grupo de compartición de colas necesario en la clase MQQUEUE o MXQUEUE. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.
- **No**: Defina un perfil hlq.NO.QUEUE.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.

6. ¿Necesita seguridad de procesos?
   - **Sí**: Active la clase MQPROC o MXPROC. Defina perfiles de proceso adecuados a nivel de gestor de colas o a nivel de grupo de compartición de colas y otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.
   - **No**: Defina un perfil hlq.NO.PROCESS.CHECKS para el gestor de colas o grupo de compartición de colas adecuado en la clase MQADMIN o MXADMIN.
7. ¿Necesita seguridad de listas de nombres?
   - **Sí**: Active la clase MQNLIST o MXNLIST. Defina los perfiles de lista de nombres adecuados a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQNLIST o MXNLIST. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.
   - **No**: Defina un perfil hlq.NO.NLIST.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
8. ¿Necesita seguridad de temas?
   - **Sí**: Active la clase MXTOPIC. Defina perfiles de tema adecuados a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MXTOPIC. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.
   - **No**: Defina un perfil hlq.NO.TOPIC.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
9. ¿Algún usuario necesita proteger el uso de las opciones `MQOPEN` o `MQPUT1` en relación con el uso de contexto?
   - **Sí**: Asegúrese de que la clase MQADMIN o MXADMIN esté activa. Defina perfiles hlq.CONTEXT.queuename a nivel de cola, de gestor de colas o de grupo de compartición de colas en la clase MQADMIN o MXADMIN. A continuación, otorgue a los usuarios o grupos adecuados permiso para acceder a estos perfiles.
   - **No**: Defina un perfil hlq.NO.CONTEXT.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
10. ¿Necesita proteger el uso de los ID de usuario alternativo?
    - **Sí**: Asegúrese de que la clase MQADMIN o MXADMIN esté activa. Defina el hlq.ALTERNATE.USER adecuado. Los perfiles *alternateuserid* para el gestor de colas o el grupo de compartición de colas necesario y otorgue a los usuarios o grupos necesarios permiso para acceder a estos perfiles.
    - **No**: Defina el perfil hlq.NO.ALTERNATE.USER.CHECKS para el gestor de colas o grupo de compartición de colas necesario en la clase MQADMIN o MXADMIN.
11. ¿Necesita personalizar los ID de usuario que se van a utilizar para las comprobaciones de seguridad de recursos mediante RESLEVEL?
    - **Sí**: Asegúrese de que la clase MQADMIN o MXADMIN esté activa. Defina un perfil hlq.RESLEVEL a nivel de gestor de colas o a nivel de grupo de compartición de colas en la clase MQADMIN o MXADMIN. A continuación otorgue a los usuarios o clases necesarios permiso para acceder al perfil.
    - **No**: Asegúrese de que no existe ningún perfil genérico en la clase MQADMIN o MXADMIN que pueda aplicarse a hlq.RESLEVEL. Defina un perfil hlq.RESLEVEL para el gestor de colas o grupo de compartición de colas necesario y asegúrese de que ningún usuario o grupo tenga acceso al mismo.
12. ¿Es necesario que los ID de usuario no utilizados de IBM MQ excedan el tiempo de espera?
    - **Sí**: Determine los valores de tiempo de espera que desea utilizar y emita el mandato MQSC ALTER SECURITY para cambiar los parámetros TIMEOUT e INTERVAL.

- **No**: Emita el mandato MQSC ALTER SECURITY para establecer el valor INTERVAL en cero.

Nota: Actualice el conjunto de datos de entrada de inicialización CSQINP1 utilizado por el subsistema para que el mandato MQSC ALTER SECURITY se emita automáticamente cuando se inicia el gestor de colas.

13. ¿Utiliza la gestión de colas distribuidas?
    - **Sí**: Utilice registros de autenticación de canal. Para obtener más información, consulte "Registros de autenticación de canal" en la página 49.
    - También puede determinar el valor de atributo MCAUSER adecuado para cada canal, o proporcionar salidas de seguridad de canal apropiadas.
14. ¿Desea utilizar TLS (seguridad de la capa de transporte)?
    - **Sí**: Para especificar que cualquier usuario que presente un certificado personal TLS que contiene un DN especificado va a utilizar un MCAUSER específico, establezca un registro de autenticación de canal de tipo SSLPEERMAP. Puede especificar un nombre distinguido individual o un patrón de caracteres que incluya comodines.
    - Planifique la infraestructura TLS. Instale la característica SSL del sistema de z/OS. En RACF, configure los filtros de nombre de certificado (CNF), si los está utilizando, y los certificados digitales. Configure el conjunto de claves SSL. Asegúrese de que el atributo SSLKEYR del gestor de colas no esté en blanco y apunte al conjunto de claves SSL. Además, asegúrese de que el valor de SSLTASKS sea como mínimo 2.
    - **No**: Asegúrese de que SSLKEYR esté en blanco y SSLTASKS sea cero.

    Para obtener información adicional acerca de TLS, consulte "Protocolos de seguridad TLS en IBM MQ" en la página 24.
15. ¿Utiliza clientes?
    - **Sí**: Utilice registros de autenticación de canal.
    - También puede determinar el valor de atributo MCAUSER adecuado para cada canal de conexión del servidor, o proporcionar salidas de seguridad de canal apropiadas, si es necesario.
16. Compruebe los valores de conmutador.

    IBM MQ emite mensajes cuando se inicia el gestor de colas que muestra los valores de seguridad. Utilice estos mensajes para determinar si los conmutadores están establecidos correctamente.
17. ¿Desea enviar contraseñas de aplicaciones cliente?
    - **Sí**: Asegúrese de que la característica z/OS está instalada y Integrated Cryptographic Service Facility (ICSF) se inicia para una mejor protección.
    - **No**: Puede ignorar el mensaje de error que indica que ICSF no se ha iniciado.

    Si desea más información sobre ISCF, consulte "Utilización de Integrated Cryptographic Service Facility (ICSF)" en la página 267

Configuración de seguridad

Esta colección de temas contiene información específica para distintos sistemas operativos y para el uso de clientes.

ULW Configuración de la seguridad en UNIX, Linux, and Windows

Consideraciones de seguridad específicas a sistemas UNIX, Linux, and Windows.

Los gestores de colas de IBM MQ transfieren información que puede ser muy valiosa, por lo que necesita utilizar un sistema de autorización para asegurar que los usuarios no autorizados no puedan acceder a sus gestores de colas. Contemple los siguientes tipos de controles de seguridad:

Quién puede administrar IBM MQ
Puede definir el conjunto de usuarios que puede emitir mandatos para administrar IBM MQ.

Quién puede utilizar objetos IBM MQ
Puede definir qué usuarios (generalmente aplicaciones) pueden utilizar llamadas MQI y mandatos PCF para realizar lo siguiente:

- Quién puede conectarse a un gestor de colas.
- Quién puede acceder a objetos (colas, definiciones de proceso, listas de nombres, canales, canales de conexión de cliente, escuchas, servicios, procesos y objetos de información de autenticación) y qué tipos de acceso tienen a dichos objetos.
- Quién puede acceder a mensajes de IBM MQ.
- Quién puede acceder a la información de contexto asociada a un mensaje.

Seguridad de canal
Debe asegurarse de que los canales que se utilizan para enviar mensajes a sistemas remotos puedan acceder a los recursos necesarios.

Puede utilizar los recursos operativos estándar para otorgar acceso a las bibliotecas de programa, las bibliotecas de enlaces de la MQI y a los mandatos. Sin embargo, el directorio que contiene las colas y otros datos de los gestores de colas es privado para IBM MQ; no utilice mandatos estándar del sistema operativo para otorgar o revocar autorizaciones a los recursos de la MQI.

ULW Cómo funcionan las autorizaciones en UNIX, Linux, and Windows

Las tablas de especificación de autorizaciones de los temas de esta sección definen de forma precisa cómo funcionan las autorizaciones y las restricciones que se aplican.

Las tablas se aplican a estas situaciones:

- Aplicaciones que emiten llamadas MQI
- Programas de administración que emiten mandatos MQSC como mandatos PCF de escape
- Programas de administración que emiten mandatos PCF

En esta sección, la información se presenta como un conjunto de tablas que especifican lo siguiente:

Acción que se va a realizar
Opción MQI, mandato MQSC o mandato PCF.

Objeto de control de acceso
Cola, proceso, gestor de colas, lista de nombres, información de autenticación, canal, canal de conexión cliente, receptor o servicio.

Autorización necesaria
Expresada como constante de tipo MQZAO_.

En las tablas, las constantes que tienen el prefijo MQZAO_ corresponden a las palabras claves de la lista de autorizaciones del mandato `setmqaut` para la entidad específica. Por ejemplo, MQZAO_BROWSE corresponde a la palabra clave `+browse`, MQZAO_SET_ALL_CONTEXT corresponde a la palabra clave `+setall`, etc. Estas constantes están definidas en el archivo de cabecera cmqzc.h que se proporciona con el producto.

ULW *Autorizaciones para llamadas MQI*

MQCONN, **MQOPEN**, **MQPUT1** y **MQCLOSE** pueden requerir comprobaciones de autorización. Las tablas de este tema muestran un resumen de las autorizaciones necesarias para cada llamada.

Una aplicación puede emitir determinadas llamadas y opciones MQI sólo si el identificador de usuario bajo el que se está ejecutando (o cuyas autorizaciones puede asumir) tiene la autorización pertinente.

Hay cuatro llamadas MQI que pueden requerir comprobaciones de autorización: **MQCONN**, **MQOPEN**, **MQPUT1** y **MQCLOSE**.

Para **MQOPEN** y **MQPUT1**, la comprobación de autorización se efectúa en el nombre del objeto que se está abriendo, y no en el nombre o nombres resultantes de la resolución del nombre. Por ejemplo, una aplicación puede tener autorización para abrir una cola alias sin tener autorización para abrir la cola base

en la que se resuelve la cola alias. La regla es que la comprobación se lleva a cabo en la primera definición encontrada durante el proceso de resolución de un nombre que no es un alias de gestor de colas, a menos que la definición de alias de gestor de colas se abra directamente; es decir, su nombre se visualiza en el campo *ObjectName* del descriptor de objeto. Para el objeto que se va a abrir siempre es necesario tener autorización. En algunos casos, también se necesita una autorización adicional independiente de la cola que se obtiene a través de una autorización para el objeto de gestor de colas.

Tabla 10 en la página 130, Tabla 11 en la página 130, Tabla 12 en la página 131 y Tabla 13 en la página 131 resumen las autorizaciones necesarias para cada llamada. La indicación *No es aplicable* significa que la comprobación de autorización no está asociada a esta operación. La indicación *No se comprueba* significa que no se realiza una comprobación de autorización.

Nota: En estas tablas no se mencionan listas de nombres, canales, canales de conexión de cliente, escuchas, servicios u objetos de información de autenticación. Esto se debe a que ninguna de las autorizaciones se aplica a estos objetos, salvo MQOO_INQUIRE, para la que se aplican las mismas autorizaciones que para los demás objetos.

La autorización especial MQZAO_ALL_MQI incluye todas las autorizaciones de las tablas que sean relevantes al tipo de objeto, excepto MQZAO_DELETE y MQZAO_DISPLAY, que están clasificadas como autorizaciones de administración.

Para poder modificar cualquier opción de contexto de mensaje, debe tener las autorizaciones apropiadas para emitir la llamada. Por ejemplo, para poder utilizar MQOO_SET_IDENTITY_CONTEXT o MQPMO_SET_IDENTITY_CONTEXT, debe tener el permiso `+setid`.

Tabla 10. Autorización de seguridad necesaria para llamadas MQCONN

Autorización necesaria para:	**Objeto de cola (“1” en la página 132)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCONN	No aplicable	No aplicable	MQZAO_CONNECT

Tabla 11. Autorización de seguridad necesaria para llamadas MQOPEN

Autorización necesaria para:	**Objeto de cola (“1” en la página 132)**	**Objeto de proceso**	**Objeto gestor de colas**
MQOO_INQUIRE	MQZAO_INQUIRE	MQZAO_INQUIRE	MQZAO_INQUIRE
MQOO_BROWSE	MQZAO_BROWSE	No aplicable	No se comprueba
MQOO_INPUT_*	MQZAO_INPUT	No aplicable	No se comprueba
MQOO_SAVE_ ALL_CONTEXT **(“2” en la página 132)**	MQZAO_INPUT	No aplicable	No aplicable
MQOO_OUTPUT (Cola normal) **(“3” en la página 132)**	MQZAO_OUTPUT	No aplicable	No aplicable
MQOO_PASS_ IDENTITY_CONTEXT **(“4” en la página 132)**	MQZAO_PASS_ IDENTITY_CONTEXT	No aplicable	No se comprueba
MQOO_PASS_ALL_ CONTEXT **(“4” en la página 132, “5” en la página 132)**	MQZAO_PASS _ALL_CONTEXT	No aplicable	No se comprueba
MQOO_SET_ IDENTITY_CONTEXT **(“4” en la página 132, “5” en la página 132)**	MQZAO_SET_ IDENTITY_CONTEXT	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“6” en la página 132)**

Tabla 11. Autorización de seguridad necesaria para llamadas MQOPEN (continuación)			
Autorización necesaria para:	**Objeto de cola (“1” en la página 132)**	**Objeto de proceso**	**Objeto gestor de colas**
MQOO_SET_ ALL_CONTEXT **(“4” en la página 132, “7” en la página 132)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 132)**
MQOO_OUTPUT (cola de transmisión) **(“8” en la página 132)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 132)**
MQOO_SET	MQZAO_SET	No aplicable	No se comprueba
MQOO_ALTERNATE_ USER_AUTHORITY	**(“9” en la página 132)**	**(“9” en la página 132)**	MQZAO_ALTERNATE_ USER_AUTHORITY **(“9” en la página 132, “10” en la página 132)**

Tabla 12. Autorización de seguridad necesaria para llamadas MQPUT1			
Autorización necesaria para:	**Objeto de cola (“1” en la página 132)**	**Objeto de proceso**	**Objeto gestor de colas**
MQPMO_PASS_ IDENTITY_CONTEXT	MQZAO_PASS_ IDENTITY_CONTEXT **(“11” en la página 132)**	No aplicable	No se comprueba
MQPMO_PASS_ALL _CONTEXT	MQZAO_PASS_ ALL_CONTEXT **(“11” en la página 132)**	No aplicable	No se comprueba
MQPMO_SET_ IDENTITY_CONTEXT	MQZAO_SET_ IDENTITY_CONTEXT **(“11” en la página 132)**	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“6” en la página 132)**
MQPMO_SET_ ALL_CONTEXT	MQZAO_SET_ ALL_CONTEXT **(“11” en la página 132)**	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 132)**
(Cola de transmisión) **(“8” en la página 132)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 132)**
MQPMO_ALTERNATE_ USER_AUTHORITY	**(“12” en la página 132)**	No aplicable	MQZAO_ALTERNATE_ USER_AUTHORITY **(“10” en la página 132)**

Tabla 13. Autorización de seguridad necesaria para llamadas MQCLOSE			
Autorización necesaria para:	**Objeto de cola (“1” en la página 132)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCO_DELETE	MQZAO_DELETE **(“13” en la página 132)**	No aplicable	No aplicable
MQCO_DELETE _PURGE	MQZAO_DELETE **(“13” en la página 132)**	No aplicable	No aplicable

Notas para las tablas:

1. Si se está abriendo una cola modelo:
   - Para la cola modelo, es necesaria la autorización MQZAO_DISPLAY además de la autorización para abrir la cola modelo correspondiente al tipo de acceso para el que se está efectuando la apertura.
   - La autorización MQZAO_CREATE no es necesaria para crear la cola dinámica.
   - El identificador de usuario utilizado para abrir la cola modelo se otorga automáticamente a todas las autorizaciones específicas de la cola (equivalentes a MQZAO_ALL) para la cola dinámica creada.
2. También debe especificarse MQOO_INPUT_*. Esto es válido para una cola local, modelo o alias.
3. Esta comprobación se realiza en todas las salidas, excepto en las colas de transmisión (vea la nota "8" en la página 132).
4. También debe especificarse MQOO_OUTPUT.
5. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT.
6. Esta autorización es necesaria tanto para el objeto gestor de colas como para la cola concreta.
7. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT y MQOO_SET_IDENTITY_CONTEXT.
8. Esta comprobación se realiza para una cola local o modelo cuyo atributo de cola *Usage* sea MQUS_TRANSMISSION y se esté abriendo directamente para salida. Esto no es aplicable si se abre una cola remota (especificando los nombres del gestor de colas remoto y la cola remota, o especificando el nombre de una definición local de la cola remota).
9. También debe especificarse como mínimo una de las opciones MQOO_INQUIRE (para cualquier tipo de objeto) o MQOO_BROWSE, MQOO_INPUT_*, MQOO_OUTPUT o MQOO_SET (para las colas). La comprobación que se lleva a cabo es la misma que en las otras opciones especificadas, utilizando el identificador de usuario alternativo suministrado para la autorización sobre el objeto específico nombrado, y la autorización sobre la aplicación actual para la comprobación MQZAO_ALTERNATE_USER_IDENTIFIER.
10. Esta autorización permite especificar cualquier *AlternateUserId*.
11. También se realiza una comprobación MQZAO_OUTPUT si la cola no tiene un atributo de cola *Usage* de MQUS_TRANSMISSION.
12. La comprobación que se lleva a cabo es la misma que en la otras opciones especificadas, utilizando el identificador de usuario alternativo suministrado para la autorización sobre la cola específica nombrada, y la autorización sobre la aplicación actual para la comprobación MQZAO_ALTERNATE_USER_IDENTIFIER.
13. La comprobación solo se lleva a cabo si se cumplen las dos sentencias siguientes:
    - Se está cerrando y suprimiendo una cola dinámica permanente.
    - La cola no la ha creado la llamada MQOPEN que ha devuelto el descriptor de objeto usado.

    De lo contrario, no hay comprobación.

ULW *Autorizaciones para mandatos MQSC en los PCF de escape*

Esta información resume las autorizaciones necesarias para cada mandato MQSC contenido en un PCF de escape.

No es aplicable significa que esta operación no tiene sentido en este tipo de objeto.

El ID de usuario bajo el que se ejecuta el programa que envía el mandato también debe tener las autorizaciones siguientes:

- Autorización MQZAO_CONNECT para el gestor de colas
- Autorización MQZAO_DISPLAY sobre el gestor de colas para realizar mandatos PCF
- Autorización para emitir los mandatos MQSC en el texto del mandato PCF de Escape

ALTER *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	MQZAO_CHANGE
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

CLEAR *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CLEAR
Tema	MQZAO_CLEAR
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	No aplicable
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable
Información de comunicación	No aplicable

DEFINE *objeto* NOREPLACE (“1” en la página 137)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(“2” en la página 137)**
Tema	MQZAO_CREATE **(“2” en la página 137)**
Proceso	MQZAO_CREATE **(“2” en la página 137)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(“2” en la página 137)**
Información de autenticación	MQZAO_CREATE **(“2” en la página 137)**
Canal	MQZAO_CREATE **(“2” en la página 137)**

Objeto	Autorización necesaria
Canal de conexión de cliente	MQZAO_CREATE **(“2” en la página 137)**
Escucha	MQZAO_CREATE **(“2” en la página 137)**
Servicio	MQZAO_CREATE **(“2” en la página 137)**
Información de comunicación	MQZAO_CREATE **(“2” en la página 137)**

DEFINE *objeto* REPLACE (“1” en la página 137, “3” en la página 137)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

DELETE *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DELETE
Tema	MQZAO_DELETE
Proceso	MQZAO_DELETE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_DELETE
Información de autenticación	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexión de cliente	MQZAO_DELETE
Escucha	MQZAO_DELETE
Servicio	MQZAO_DELETE
Información de comunicación	MQZAO_DELETE

DISPLAY *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DISPLAY
Tema	MQZAO_DISPLAY

Objeto	Autorización necesaria
Proceso	MQZAO_DISPLAY
Gestor de colas	MQZAO_DISPLAY
Lista de nombres	MQZAO_DISPLAY
Información de autenticación	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexión de cliente	MQZAO_DISPLAY
Escucha	MQZAO_DISPLAY
Servicio	MQZAO_DISPLAY
Información de comunicación	MQZAO_DISPLAY

START *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL
Información de comunicación	No aplicable

STOP *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL

Objeto	Autorización necesaria
Información de comunicación	No aplicable

Mandatos de canal

Mandato	Objeto	Autorización necesaria
PING CHANNEL	Canal	MQZAO_CONTROL
RESET CHANNEL	Canal	MQZAO_CONTROL_EXTENDED
RESOLVE CHANNEL	Canal	MQZAO_CONTROL_EXTENDED

Mandatos de suscripción

Mandato	Objeto	Autorización necesaria
ALTER SUB	Tema	MQZAO_CONTROL
DEFINE SUB	Tema	MQZAO_CONTROL
DELETE SUB	Tema	MQZAO_CONTROL
DISPLAY SUB	Tema	MQZAO_DISPLAY

Mandatos de seguridad

Mandato	Objeto	Autorización necesaria
SET AUTHREC	Gestor de colas	MQZAO_CHANGE
DELETE AUTHREC	Gestor de colas	MQZAO_CHANGE
DISPLAY AUTHREC	Gestor de colas	MQZAO_DISPLAY
DISPLAY AUTHSERV	Gestor de colas	MQZAO_DISPLAY
DISPLAY ENTAUTH	Gestor de colas	MQZAO_DISPLAY
SET CHLAUTH	Gestor de colas	MQZAO_CHANGE
DISPLAY CHLAUTH	Gestor de colas	MQZAO_DISPLAY
REFRESH SECURITY	Gestor de colas	MQZAO_CHANGE

Muestra el estado

Mandato	Objeto	Autorización necesaria
DISPLAY CHSTATUS	Gestor de colas	MQZAO_DISPLAY Tenga en cuenta que la autorización `+inq` (o equivalente MQZAO_INQUIRE) es necesaria en la cola de transmisión si el tipo de canal es CLUSSDR.
DISPLAY LSSTATUS	Gestor de colas	MQZAO_DISPLAY
DISPLAY PUBSUB	Gestor de colas	MQZAO_DISPLAY
DISPLAY SBSTATUS	Gestor de colas	MQZAO_DISPLAY
DISPLAY SVSTATUS	Gestor de colas	MQZAO_DISPLAY

Mandato	Objeto	Autorización necesaria
DISPLAY TPSTATUS	Gestor de colas	MQZAO_DISPLAY

Mandatos de clúster

Mandato	Objeto	Autorización necesaria
DISPLAY CLUSQMGR	Gestor de colas	MQZAO_DISPLAY
REFRESH CLUSTER	grupo de pertenencia 'mqm' necesario	
RESET CLUSTER	grupo de pertenencia 'mqm' necesario	
SUSPEND QMGR	grupo de pertenencia 'mqm' necesario	
RESUME QMGR	grupo de pertenencia 'mqm' necesario	

Otros mandatos administrativos

Mandato	Objeto	Autorización necesaria
PING QMGR	Gestor de colas	MQZAO_DISPLAY
REFRESH QMGR	Gestor de colas	MQZAO_CHANGE
RESET QMGR	Gestor de colas	MQZAO_CHANGE
DISPLAY CONN	Gestor de colas	MQZAO_DISPLAY
STOP CONN	Gestor de colas	MQZAO_CHANGE

Nota:

1. Para los mandatos DEFINE, se necesita también la autorización MQZAO_DISPLAY sobre el objeto LIKE, si se ha especificado uno, o sobre el objeto SYSTEM.DEFAULT.xxx adecuado si se ha omitido LIKE.
2. La autorización MQZAO_CREATE no es específica de un objeto o tipo de objeto en particular. Para un gestor de colas especificado, la autorización de creación se otorga para todos los objetos, especificando un tipo de objeto QMGR en el mandato `setmqaut`.
3. Esto es aplicable si el objeto que debe sustituirse ya existe. Si no existe, la comprobación es como para DEFINE *objeto* NOREPLACE.

Información relacionada

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

ULW *Autorizaciones para mandatos PCF*

Esta sección resume las autorizaciones necesarias para cada mandato PCF.

La indicación *No se comprueba* significa que no se lleva a cabo ninguna comprobación de autorización; *No aplicable* significa que esta operación no es relevante para este tipo de objeto.

El ID de usuario bajo el que se ejecuta el programa que envía el mandato también debe tener las autorizaciones siguientes:

- Autorización MQZAO_CONNECT para el gestor de colas
- Autorización MQZAO_DISPLAY sobre el gestor de colas para realizar mandatos PCF

La autorización especial MQZAO_ALL_ADMIN incluye todas las autorizaciones de la lista siguiente que sean relevantes para el tipo de objeto, excepto MQZAO_CREATE, que no es específica de un objeto o tipo de objeto determinado.

Change *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	MQZAO_CHANGE
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

Clear *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CLEAR
Tema	MQZAO_CLEAR
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	No aplicable
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable
Información de comunicación	No aplicable

Copy *objeto* (sin sustitución) (1)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(2)**
Tema	MQZAO_CREATE **(2)**
Proceso	MQZAO_CREATE **(2)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(2)**
Información de autenticación	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**

Objeto	Autorización necesaria
Canal de conexión de cliente	MQZAO_CREATE **(2)**
Escucha	MQZAO_CREATE **(2)**
Servicio	MQZAO_CREATE **(2)**
Información de comunicación	MQZAO_CREATE **(“2” en la página 144)**

Copiar *objeto* (con sustitución) (1, 4)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

Create *objeto* (sin sustitución) (3)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(2)**
Tema	MQZAO_CREATE **(2)**
Proceso	MQZAO_CREATE **(2)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(2)**
Información de autenticación	MQZAO_CREATE **(2)**
Canal	MQZAO_CREATE **(2)**
Canal de conexión de cliente	MQZAO_CREATE **(2)**
Escucha	MQZAO_CREATE **(2)**
Servicio	MQZAO_CREATE **(2)**
Información de comunicación	MQZAO_CREATE **(2)**

Crear *objeto* (con sustitución) (3, 4)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE

Objeto	Autorización necesaria
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

Delete *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DELETE
Tema	MQZAO_DELETE
Proceso	MQZAO_DELETE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_DELETE
Información de autenticación	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexión de cliente	MQZAO_DELETE
Escucha	MQZAO_DELETE
Servicio	MQZAO_DELETE
Información de comunicación	MQZAO_DELETE

Inquire *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DISPLAY
Tema	MQZAO_DISPLAY
Proceso	MQZAO_DISPLAY
Gestor de colas	MQZAO_DISPLAY
Lista de nombres	MQZAO_DISPLAY
Información de autenticación	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexión de cliente	MQZAO_DISPLAY
Escucha	MQZAO_DISPLAY
Servicio	MQZAO_DISPLAY

Objeto	Autorización necesaria
Información de comunicación	MQZAO_DISPLAY

Inquire *objeto* names

Objeto	Autorización necesaria
Cola	No se comprueba
Tema	No se comprueba
Proceso	No se comprueba
Gestor de colas	No se comprueba
Lista de nombres	No se comprueba
Información de autenticación	No se comprueba
Canal	No se comprueba
Canal de conexión de cliente	No se comprueba
Escucha	No se comprueba
Servicio	No se comprueba
Información de comunicación	No se comprueba

Inicie *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL
Información de comunicación	No aplicable

Pare *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable

Objeto	Autorización necesaria
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL
Información de comunicación	No aplicable

Mandatos de canal

Mandato	Objeto	Autorización necesaria
Sondear canal	Canal	MQZAO_CONTROL
Restablecer canal	Canal	MQZAO_CONTROL_EXTENDED
Resolver canal	Canal	MQZAO_CONTROL_EXTENDED

Mandatos de suscripción

Mandato	Objeto	Autorización necesaria
Cambiar suscripción	Tema	MQZAO_CONTROL
Crear suscripción	Tema	MQZAO_CONTROL
Suprimir suscripción	Tema	MQZAO_CONTROL
Consultar suscripción	Tema	MQZAO_DISPLAY

Mandatos de seguridad

Mandato	Objeto	Autorización necesaria
Establecer registro de autorización	Gestor de colas	MQZAO_CHANGE
Suprimir registro de autorización	Gestor de colas	MQZAO_CHANGE
Consultar registros de autorización	Gestor de colas	MQZAO_DISPLAY
Consultar servicio de autorización	Gestor de colas	MQZAO_DISPLAY
Consultar autorización de entidad	Gestor de colas	MQZAO_DISPLAY
Establecer registro de autenticación de canal	Gestor de colas	MQZAO_CHANGE
Consultar registros de autenticación de canal	Gestor de colas	MQZAO_DISPLAY
Renovar seguridad	Gestor de colas	MQZAO_CHANGE

Muestra el estado

Mandato	Objeto	Autorización necesaria
Consultar estado del canal	Gestor de colas	MQZAO_DISPLAY Tenga en cuenta que la autorización `+inq` (o equivalente MQZAO_INQUIRE) es necesaria en la cola de transmisión si el tipo de canal es CLUSSDR.
Consultar estado de escucha de canal	Gestor de colas	MQZAO_DISPLAY
Consultar estado de publicación/suscripción	Gestor de colas	MQZAO_DISPLAY
Consultar estado de suscripción	Gestor de colas	MQZAO_DISPLAY
Consultar estado del servicio	Gestor de colas	MQZAO_DISPLAY
Consultar estado de tema	Gestor de colas	MQZAO_DISPLAY

Mandatos de clúster

Mandato	Objeto	Autorización necesaria
Consultar gestor de colas de clúster	Gestor de colas	MQZAO_DISPLAY
Renovar clúster	grupo de pertenencia 'mqm' necesario	grupo de pertenencia 'mqm' necesario
Restablecer clúster	grupo de pertenencia 'mqm' necesario	grupo de pertenencia 'mqm' necesario
Suspender clúster de gestores de colas	grupo de pertenencia 'mqm' necesario	grupo de pertenencia 'mqm' necesario
Reanudar clúster de gestores de colas	grupo de pertenencia 'mqm' necesario	grupo de pertenencia 'mqm' necesario

Otros mandatos administrativos

Mandato	Objeto	Autorización necesaria
Sondear gestor de colas	Gestor de colas	MQZAO_DISPLAY
Renovar gestor de colas	Gestor de colas	MQZAO_CHANGE
Restablecer gestor de colas	Gestor de colas	MQZAO_CHANGE
Restablecer estadísticas de la cola	Cola	MQZAO_DISPLAY y MQZAO_CHANGE
Consultar conexión	Gestor de colas	MQZAO_DISPLAY
Detener conexión	Gestor de colas	MQZAO_CHANGE

Nota:

1. En los mandatos Copy, también es necesaria la autorización MQZAO_DISPLAY para el objeto de origen.

2. La autorización MQZAO_CREATE no es específica de un objeto o tipo de objeto en particular. Para un gestor de colas especificado, la autorización de creación se otorga para todos los objetos, especificando un tipo de objeto QMGR en el mandato `setmqaut`.
3. Para los mandatos Create, también se necesita la autorización MQZAO_DISPLAY para el SYSTEM.DEFAULT.* .
4. Esto es aplicable si el objeto que debe sustituirse ya existe. Si no existe, la comprobación es como para un mandato Copy o Create sin sustitución.

AIX Creación y gestión de grupos en AIX

En AIX, siempre y cuando no esté utilizando NIS o NIS+, utilice SMITTY para trabajar con grupos.

Acerca de esta tarea

En AIX, puede utilizar SMITTY para crear un grupo, añadir un usuario a un grupo, mostrar una lista de los usuarios que están en el grupo y eliminar un usuario de un grupo.

Procedimiento

1. Desde SMITTY, seleccione **Seguridad y usuarios** y pulse Intro.
2. Seleccione **Grupos** y pulse Intro.
3. Para crear un grupo, realice los pasos siguientes:
   a) Seleccione **Añadir un grupo** y pulse Intro.
   b) Escriba el nombre del grupo y los nombres de los usuarios que desee añadir al grupo, separados por comas.
   c) Pulse Intro para crear el grupo.
4. Para añadir un usuario a un grupo, efectúe los pasos siguientes:
   a) Seleccione **Cambiar/Mostrar características de grupos** y pulse Intro.
   b) Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.
   c) Añada los nombres de los usuarios que desea añadir al grupo, separados por comas.
   d) Pulse Intro para añadir los nombres al grupo.
5. Para mostrar quién está en un grupo, realice los pasos siguientes:
   a) Seleccione **Cambiar/Mostrar características de grupos** y pulse Intro.
   b) Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.
6. Para eliminar un usuario de un grupo, realice los pasos siguientes:
   a) Seleccione **Cambiar/Mostrar características de grupos** y pulse Intro.
   b) Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.
   c) Suprima el nombre del usuario que desea eliminar del grupo.
   d) Pulse Intro para eliminar el nombre del grupo.

Linux Creación y gestión de grupos en Linux

En Linux, siempre y cuando no esté utilizando NIS o NIS+, utilice el archivo `/etc/group` para trabajar con grupos.

Acerca de esta tarea

En Linux, la información de grupo se mantiene en el archivo `/etc/group`. Puede utilizar mandatos para crear grupos, añadir usuarios a grupos, visualizar una lista de los usuarios que están en un grupo y eliminar usuarios de un grupo.

Procedimiento

1. Para crear un nuevo grupo, utilice el mandato **groupadd**.
   Escriba el siguiente mandato:

   ```
   groupadd -g group-ID group-name
   ```

   donde *ID_grupo* es el identificador numérico del grupo y *nombre_grupo* es el nombre del grupo.
2. Para añadir un miembro a un grupo adicional, utilice el mandato **usermod** que enumera los grupos adicionales de los que el usuario es miembro actualmente y los grupos adicionales de los que el usuario va a ser miembro.
   Por ejemplo, si el usuario ya es miembro del grupo `groupa` y va a convertirse en miembro de `groupb`, utilice el mandato siguiente:

   ```
   usermod -G groupa,groupb user-name
   ```

   donde *nombre_usuario* es el nombre de usuario.
3. Para visualizar los miembros de un grupo, utilice el mandato **getent**.
   Escriba el siguiente mandato:

   ```
   getent group group-name
   ```

   donde *nombre-grupo* es el nombre del grupo.
4. Para eliminar un miembro de un grupo adicional, utilice el mandato **usermod**, que enumera los grupos adicionales de los que desea que el usuario siga siendo miembro.
   Por ejemplo, si el grupo primario del usuario es `users` y el usuario también es miembro de los grupos `mqm`, `groupa` y `groupb`, para eliminar el usuario del grupo `mqm`, utilice el mandato siguiente:

   ```
   usermod -G groupa,groupb user-name
   ```

   donde *nombre_usuario* es el nombre de usuario.

Solaris Creación y gestión de grupos en Solaris

En Solaris, siempre y cuando no esté utilizando NIS o NIS+, utilice el archivo `/etc/group` para trabajar con grupos.

Acerca de esta tarea

En Solaris, la información de grupo se mantiene en el archivo `/etc/group`. Puede utilizar mandatos para crear grupos, añadir usuarios a grupos, visualizar una lista de los usuarios que están en un grupo y eliminar usuarios de un grupo.

Procedimiento

1. Para crear un nuevo grupo, utilice el mandato **groupadd**.
   Escriba el siguiente mandato:

   ```
   groupadd -g group-ID group-name
   ```

   donde *ID_grupo* es el identificador numérico del grupo y *nombre_grupo* es el nombre del grupo.
2. Para añadir un miembro a un grupo adicional, utilice el mandato **usermod** que enumera los grupos adicionales de los que el usuario es miembro actualmente y los grupos adicionales de los que el usuario va a ser miembro.
   Por ejemplo, si el usuario ya es miembro del grupo `groupa` y va a convertirse en miembro de `groupb`, utilice el mandato siguiente:

   ```
   usermod -G groupa,groupb user-name
   ```

donde *nombre_usuario* es el nombre de usuario.

3. Para saber quién es miembro de un grupo, consulte la entrada de ese grupo en el archivo `/etc/group`.
4. Para eliminar un miembro de un grupo adicional, utilice el mandato **usermod**, que enumera los grupos adicionales de los que desea que el usuario siga siendo miembro.
   Por ejemplo, si el grupo primario del usuario es `users` y el usuario también es miembro de los grupos `mqm`, `groupa` y `groupb`, para eliminar el usuario del grupo `mqm`, utilice el mandato siguiente:

   ```
   usermod -G groupa,groupb user-name
   ```

   donde *nombre_usuario* es el nombre de usuario.

Windows Creación y gestión de grupos en Windows

En Windows, utilice la característica Administración de equipos para administrar los grupos en una estación de trabajo o en una máquina del servidor miembro.

Acerca de esta tarea

Para los controladores de dominio, los usuarios y grupos se administran mediante Active Directory. Para obtener más información sobre la utilización de Active Directory, consulte las instrucciones correspondientes del sistema operativo.

Los cambios que realice en la pertenencia a un grupo principal no se reconocen hasta que se reinicia el gestor de colas o se emite el mandato MQSC **REFRESH SECURITY** (o el equivalente PCF).

Utilice el panel Administración de equipos de Windows para trabajar con el usuario y los grupos. Puede que los cambios efectuados en la sesión iniciada actual no sean efectivos hasta que se vuelva a iniciar la sesión.

Windows *Creación de un grupo en Windows*

Crear un grupo utilizando el panel de control.

Procedimiento

1. Abra el Panel de control
2. Efectúe una doble pulsación en **Herramientas administrativas**.
   Se abre el panel Herramientas administrativas.
3. Efectúe una doble pulsación en **Administración de equipos**.
   Se abre el panel Administración de equipos.
4. Expanda **Usuarios locales y grupos**.
5. Pulse el botón derecho del ratón en **Grupos** y seleccione **Grupo nuevo...**
   Aparece el panel Grupo nuevo.
6. Escriba un nombre adecuado en el campo Nombre de grupo y pulse **Crear**.
7. Pulse **Cerrar**.

Windows *Adición de un usuario a un grupo en Windows*

Añada un usuario a un grupo utilizando el panel de control.

Procedimiento

1. Abra el Panel de control
2. Efectúe una doble pulsación en **Herramientas administrativas**.
   Se abre el panel Herramientas administrativas.
3. Efectúe una doble pulsación en **Administración de equipos**.

Se abre el panel Administración de equipos.

4. Desde el panel Administración de equipos, expanda **Usuarios locales y grupos**.
5. Seleccione **Usuarios**
6. Efectúe una doble pulsación en el usuario que desea añadir a un grupo.
   Aparece el panel de propiedades de usuario.
7. Seleccione el separador **Miembro de**.
8. Seleccione el grupo al que desea añadir el usuario. Si el grupo que desea no está visible:
   a) Pulse **Añadir...**.
      Aparece el panel Seleccionar grupos.
   b) Pulse **Ubicaciones...**.
      Aparece el panel Ubicaciones.
   c) Seleccione la ubicación del grupo al que desea añadir el usuario en la lista y pulse **Aceptar**.
   d) Escriba el nombre de grupo en el campo correspondiente.

      De forma alternativa, pulse **Avanzado ...** y, a continuación, **Buscar ahora** para listar los grupos disponibles en la ubicación seleccionada actualmente. Aquí, seleccione el grupo al que desea añadir el usuario y pulse **Aceptar**.
   e) Pulse **Aceptar**.
      Aparece el panel de propiedades de usuario, que muestra el grupo que ha añadido.
   f) Seleccione el grupo.
9. Pulse **Aceptar**.
   Aparece el panel Administración de equipos.

Windows *Visualización de los miembros de un grupo en Windows*

Mostrar los miembros de un grupo utilizando el panel de control.

Procedimiento

1. Abra el Panel de control
2. Efectúe una doble pulsación en **Herramientas administrativas**.
   Se abre el panel Herramientas administrativas.
3. Efectúe una doble pulsación en **Administración de equipos**.
   Se abre el panel Administración de equipos.
4. Desde el panel Administración de equipos, expanda **Usuarios locales y grupos**.
5. Seleccione **Grupos**.
6. Efectúe una doble pulsación en un grupo. Aparece el panel de propiedades del grupo.
   Aparece el panel de propiedades del grupo.

Resultados

Se muestran los miembros del grupo.

Windows *Supresión de un usuario de un grupo en Windows*

Eliminar un usuario de un grupo utilizando el panel de control.

Procedimiento

1. Abra el Panel de control
2. Efectúe una doble pulsación en **Herramientas administrativas**.
   Se abre el panel Herramientas administrativas.

3. Efectúe una doble pulsación en **Administración de equipos**.
   Se abre el panel Administración de equipos.
4. Desde el panel Administración de equipos, expanda **Usuarios locales y grupos**.
5. Seleccione **Usuarios**.
6. Efectúe una doble pulsación en el usuario que desea añadir a un grupo.
   Aparece el panel de propiedades de usuario.
7. Seleccione el separador **Miembro de**.
8. Seleccione el grupo del que desea eliminar el usuario y luego pulse **Quitar**.
9. Pulse **Aceptar**.
   Aparece el panel Administración de equipos.

Resultados

Ya ha eliminado al usuario del grupo.

Windows Consideraciones especiales de seguridad en Windows

Algunas funciones de seguridad se comportan de forma diferente en las distintas versiones de Windows.

La seguridad de IBM MQ depende de llamadas a la API del sistema operativo para obtener información sobre autorizaciones de usuario y pertenencias a grupos. Algunas funciones no se comportan del mismo modo en los sistemas Windows. Esta colección de temas incluye descripciones de cómo estas diferencias pueden afectar a la seguridad de IBM MQ cuando se ejecuta IBM MQ en un entorno Windows.

Windows *Cuentas de usuario local y de dominio para el servicio IBM MQ Windows*

Cuando IBM MQ está en ejecución, debe comprobar que sólo los usuarios autorizados pueden acceder a los gestores de colas o a las colas. Esto requiere una cuenta de usuario especial que IBM MQ puede utilizar para consultar información sobre el usuario que intenta dicho acceso.

- "Configuración de cuentas de usuario especiales con el Prepare IBM MQ Wizard" en la página 148
- "Utilización de IBM MQ con Active Directory" en la página 149
- "Derechos de usuario necesarios para un servicio IBM MQ Windows" en la página 149

Configuración de cuentas de usuario especiales con el Prepare IBM MQ Wizard

El Prepare IBM MQ Wizard crea una cuenta de usuario especial para que se los procesos que tengan que utilizar el servicio Windows puedan compartirlo (consulte Configuración de IBM MQ con el asistente de preparación de IBM MQ).

Un servicio de Windows se comparte entre procesos de cliente para una instalación de IBM MQ. Se crea un servicio para cada instalación. Cada servicio se denomina `MQ_`*`InstallationName`* y tiene un nombre de visualización de `IBM MQ(`*`InstallationName`*`)`.

Puesto que cada servicio se debe compartir entre sesiones de inicio de sesión interactivas y no interactivas, debe iniciar cada una bajo una cuenta de usuario especial. Puede utilizar una cuenta de usuario especial para todos los servicios o crear distintas cuentas de usuario especiales. Cada cuenta de usuario especial debe tener el derecho de usuario `Iniciar sesión como servicio`; para obtener más información, consulte Tabla 14 en la página 149. Si el ID de usuario no tiene autorización para poder ejecutar el servicio, el servicio no se inicia y se devuelve un error al registro de sucesos del sistema de Windows. Normalmente, tendrá que ejecutar Prepare IBM MQ Wizard y configurar el ID de usuario correctamente. Sin embargo, si ha configurado el ID de usuario manualmente, es posible que pueda tener un problema que será necesario resolver.

Cuando instala IBM MQ y ejecuta el Prepare IBM MQ Wizard por primera vez, se crea una cuenta de usuario local para el servicio denominado MUSR_MQADMIN con los valores y permisos necesarios, incluido `Iniciar sesión como un servicio`.

En instalaciones posteriores, el Prepare IBM MQ Wizard creará una cuenta de usuario denominada MUSR_MQADMIN*x*, donde *x* es el siguiente número disponible que representa un ID de usuario que no existe. La contraseña para MUSR_MQADMIN*x* se genera aleatoriamente cuando se crea la cuenta y se utiliza para configurar el entorno de inicio de sesión para el servicio. La contraseña generada no caduca.

Esta cuenta de IBM MQ MQ no se ve afectada por ninguna de las políticas de cuentas definidas en el sistema que requieren que las contraseñas se cambien después de un cierto periodo de tiempo.

La contraseña no se conoce fuera de este proceso de un solo uso y la almacena el sistema operativo de Windows en una parte segura del registro.

Utilización de IBM MQ con Active Directory

En algunas configuraciones de red, donde las cuentas de usuario se definen en controladores de dominios que están utilizando el servicio de directorios Active Directory, la cuenta de usuario local bajo la cual se está ejecutando IBM MQ podría no tener la autorización que requiere para consultar la pertenencia de otras cuentas de usuario de dominio. Cuando instala IBM MQ, el Prepare IBM MQ Wizard identifica si este es el caso realizando pruebas y formulando preguntas sobre la configuración de red.

Si la cuenta de usuario local con la que se ejecuta IBM MQ no tiene la autorización necesaria, el Prepare IBM MQ Wizard le solicita los detalles de la cuenta de usuario de dominio con derechos de usuario concretos. Para obtener información acerca de cómo crear y configurar una cuenta de dominio de Windows, consulte Creación y configuración de cuentas de dominio de Windows para IBM MQ. Para ver los derechos de usuario que necesita la cuenta de usuario de dominio, consulte Tabla 14 en la página 149.

Cuando ha especificado los detalles de cuenta válidos para la cuenta de usuario de dominio en el Prepare IBM MQ Wizard, éste último configura un servicio de IBM MQWindows para que se ejecute con la nueva cuenta. Los detalles de la cuenta se guardan en una parte segura del Registro que no pueden leer los usuarios.

Cuando el servicio se ejecuta, se inicia un servicio IBM MQ Windows y permanece en ejecución mientras se ejecuta el servicio. Un administrador de IBM MQ que inicie la sesión en el servidor después de haber lanzado el servicio de Windows puede utilizar IBM MQ Explorer para administrar gestores de colas en el servidor. Esto conecta IBM MQ Explorer al proceso del servicio de Windows existente. Estas dos acciones necesitan niveles de permiso diferentes para poder funcionar:

- El proceso de inicio necesita un permiso de inicio.
- El administrador de IBM MQ requiere permiso de acceso.

Derechos de usuario necesarios para un servicio IBM MQ Windows

En la tabla siguiente se muestran los derechos de usuario necesarios para las cuentas de usuario local y de dominio con las que se ejecuta el servicio Windows para una instalación de IBM MQ.

Tabla 14. Derechos de usuario necesarios para un servicio Windows de IBM MQ

Permiso	**Descripción**
Iniciar sesión como proceso por lotes	Permite que un servicio IBM MQ Windows se ejecute en esta cuenta de usuario.
Iniciar sesión como servicio	Permite a los usuarios establecer el servicio IBM MQ Windows para iniciar sesión utilizando la cuenta configurada.
Concluir el sistema	Permite al servicio IBM MQ Windows reiniciar el servidor si está configurado para ello cuando falla la recuperación de un servicio.
Aumentar cuotas	Necesario para la llamada `CreateProcessAsUser` del sistema operativo.

Tabla 14. Derechos de usuario necesarios para un servicio Windows de IBM MQ (continuación)	
Permiso	**Descripción**
Actuar como parte del sistema operativo	Necesario para la llamada `LogonUser` del sistema operativo.
Eludir la comprobación cruzada	Necesario para la llamada `LogonUser` del sistema operativo.
Sustituir una señal de nivel de proceso	Necesario para la llamada `LogonUser` del sistema operativo.

Nota: Es posible que se necesiten derechos para depurar programas en entornos que ejecutan aplicaciones ASP e IIS.

Su cuenta de usuario de dominio debe tener estos derechos de usuario de Windows establecidos como derechos de usuario efectivos, tal como se indica en la aplicación Directiva de seguridad local. Si no lo están, establézcalos utilizando la aplicación Directiva de seguridad local localmente en el servidor, o utilizando la aplicación Directiva de seguridad de dominio para todo el dominio.

Windows *Permisos de seguridad de Windows Server*

La instalación de IBM MQ tiene un comportamiento distinto en Windows Server según si la instalación la realiza un usuario local o un usuario de dominio.

Si un usuario *local* instala IBM MQ, el Prepare IBM MQ Wizard detecta que el usuario local creado para el servicio IBM MQ Windows puede recuperar la información de pertenencia a grupos del usuario que realiza la instalación. El Prepare IBM MQ Wizard solicita al usuario información sobre la configuración de red para determinar si hay otras cuentas de usuario definidas o no en los controladores de dominio que se ejecutan en Windows 2000 o posterior. De esta forma, el servicio de IBM MQ Windows se debe ejecutar bajo una cuenta de usuario de dominio con autoridades y valores particulares. El Prepare IBM MQ Wizard solicita al usuario los detalles de la cuenta de este usuario como se describe en Configuración de IBM MQ con el asistente de preparación de IBM MQ.

Si un usuario *domain* instala IBM MQ, el Prepare IBM MQ Wizard detecta que el usuario local creado para el servicio IBM MQ Windows no puede recuperar la información de pertenencia a grupos del usuario que realiza la instalación. En este caso, el Prepare IBM MQ Wizard siempre solicita al usuario los detalles de la cuenta de usuario de dominio para que el servicio IBM MQ Windows los utilice.

Cuando el servicio IBM MQ Windows debe utilizar una cuenta de usuario de dominio, IBM MQ no puede operar correctamente hasta que esto se haya configurado utilizando el Prepare IBM MQ Wizard. El Prepare IBM MQ Wizard no permite que el usuario siga con otras tareas, hasta que no se haya configurado el servicio Windows con una cuenta adecuada.

Para obtener más información, consulte Creación y configuración de cuentas de dominio para IBM MQ.

Windows *Cambio del nombre de usuario asociado al servicio IBM MQ*

Para cambiar el nombre de usuario asociado al servicio IBM MQ cree una cuenta nueva y especifique los detalles mediante el Prepare IBM MQ Wizard.

Acerca de esta tarea

Cuando instala IBM MQ y ejecuta el Prepare IBM MQ Wizard por primera vez, se crea una cuenta de usuario local para el servicio denominada MUSR_MQADMIN. En instalaciones posteriores, el Prepare IBM MQ Wizard creará una cuenta de usuario denominada MUSR_MQADMIN*x*, donde *x* es el siguiente número disponible que representa un ID de usuario que no existe.

Es posible que tenga que cambiar el nombre de usuario asociado al servicio IBM MQ, MUSR_MQADMIN o MUSR_MQADMIN*x*, por algún otro. Por ejemplo, es posible que tenga que hacer esto si el gestor de colas está asociado a Db2, que no acepta nombres de usuario de más de 8 caracteres.

Procedimiento

1. Cree una nueva cuenta de usuario (por ejemplo, **NEW_NAME**)
2. Utilice el Prepare IBM MQ Wizard para especificar los detalles de la nueva cuenta de usuario.

Tareas relacionadas

Configuración de IBM MQ con el Asistente de preparación de IBM MQ

Windows *Cambiar la contraseña de la cuenta de usuario local del servicio IBM MQ Windows*

Puede cambiar la contraseña de la cuenta de usuario local del servicio de IBM MQ Windows utilizando el panel Administración de equipos.

Acerca de esta tarea

Para cambiar la contraseña de la cuenta de usuario local de servicio de IBM MQWindows , realice los pasos siguientes:

Procedimiento

1. Identifique el usuario en el que se ejecuta el servicio.
2. Detenga el servicio IBM MQ desde el panel Administración de equipos.
3. Cambie la contraseña necesaria igual que lo haría con una contraseña personal.
4. Vaya a las propiedades del servicio IBM MQ desde el panel Administración de equipos.
5. Seleccione la página **Iniciar sesión**.
6. Confirme que el nombre de cuenta especificado coincida con el usuario para el que se ha modificado la contraseña.
7. Escriba la contraseña en los campos **Contraseña** y **Confirmar contraseña** y pulse **Aceptar**.

Windows *Cambio de la contraseña de un servicio IBM MQ Windows para una instalación que se ejecuta con una cuenta de usuario de dominio*

Como alternativa a la utilización del Prepare IBM MQ Wizard para especificar los detalles de la cuenta de usuario de dominio, puede utilizar el panel Administración de equipos para modificar los detalles de **Iniciar sesión** para el servicio IBM MQ específico de la instalación.

Acerca de esta tarea

Si el servicio de IBM MQWindows para una instalación se está ejecutando bajo una cuenta de usuario de dominio, puede cambiar la contraseña de la cuenta de la siguiente manera:

Procedimiento

1. Cambie la contraseña de la cuenta de dominio en el controlador de dominio. Es posible que debe pedirle al administrador del sistema que realice esta tarea.
2. Complete los pasos siguientes para modificar la página **Iniciar sesión** para el servicio IBM MQ.
   a) Identifique el usuario con el que se ejecuta el servicio.
   b) Detenga el servicio IBM MQ desde el panel Administración de equipos.
   c) Cambie la contraseña necesaria igual que lo haría con una contraseña personal.
   d) Vaya a las propiedades del servicio IBM MQ desde el panel Administración de equipos.
   e) Seleccione la página **Iniciar sesión**.
   f) Confirme que el nombre de cuenta especificado coincida con el usuario para el que se ha modificado la contraseña.
   g) Escriba la contraseña en los campos **Contraseña** y **Confirmar contraseña** y pulse **Aceptar**.

La cuenta de usuario bajo la cual se ejecute el servicio IBM MQ Windows ejecuta los mandatos MQSC que han emitido las aplicaciones de la interfaz de usuario, o que se han realizado automáticamente durante el arranque del sistema, la conclusión o la recuperación del servicio. Por lo tanto, esta cuenta de usuario debe tener derechos de administración de IBM MQ. De forma predeterminada, se añade al grupo mqm local en el servidor. Si se elimina esta pertenencia, el servicio IBM MQ Windows no funciona. Para obtener más información sobre los derechos de usuario, consulte "Derechos de usuario necesarios para un servicio IBM MQ Windows" en la página 149.

Si surge un problema de seguridad con la cuenta de usuario bajo la que se ejecuta el servicio IBM MQ Windows, aparecerán mensajes de error y descripciones en la anotación de sucesos del sistema.

Tareas relacionadas

Configuración de IBM MQ con el Asistente de preparación de IBM MQ

Windows *Consideraciones sobre la promoción de servidores Windows en los controladores de dominio*

Cuando promueve un servidor Windows a un controlador de dominio, considere si el valor de seguridad relacionado con los permisos de usuario y grupo es adecuado. Al cambiar el estado de una máquina de Windows entre el servidor y el controlador de dominio, tenga en cuenta que esto puede afectar al funcionamiento de IBM MQ porque IBM MQ utiliza un grupo mqm definido localmente.

Valores de seguridad relacionados con los permisos de usuario y grupo de dominio

IBM MQ se basa en la información de pertenencia a grupos para implementar su política de seguridad, lo que significa que es importante que el ID de usuario que realiza operaciones de IBM MQ pueda determinar la pertenencia a grupos de otros usuarios.

Cuando promueve un servidor Windows a un controlador de dominio, se le presenta una opción para el valor de seguridad relacionado con los permisos de usuario y grupo. Esta opción controla si los usuarios arbitrarios pueden recuperar miembros de grupo desde Active Directory. Si se configura un controlador de dominio para que las cuentas locales tengan autorización para consultar la pertenencia a grupos de las cuentas de usuario de dominio, el ID de usuario predeterminado creado por IBM MQ durante el proceso de instalación puede obtener la pertenencia a grupos de otros usuarios según sea necesario. Sin embargo, si se configura un controlador de dominio de modo que las cuentas locales no tengan autorización para consultar la pertenencia a grupos de las cuentas de usuario de dominio, esto impide que IBM MQ complete sus comprobaciones de que los usuarios definidos en el dominio tengan autorización para acceder a los gestores de colas o las colas, y el acceso falla. Si está utilizando Windows en un controlador de dominio que se ha configurado de esta forma, se debe utilizar una cuenta de usuario de dominio especial con los permisos necesarios.

En este caso, debe saber:

- Cómo se comportan los permisos de seguridad para la versión de Windows.
- Cómo permitir que los miembros del grupo mqm de dominio lean la información de pertenencia a grupos.
- Cómo configurar un servicio de IBM MQWindows para que se ejecute bajo un usuario de dominio.

Para obtener más información, consulte Configuración de cuentas de usuario de IBM MQ.

Acceso de IBM MQ al grupo mqm local

Cuando los servidores Windows se promocionan o se degradan en controladores de dominio, IBM MQ pierde el acceso al grupo mqm local.

Cuando un servidor se promociona para que sea un controlador de dominio, el ámbito cambia de local a local del dominio. Cuando la máquina se degrada a servidor, todos los grupos locales del dominio se eliminan. Esto significa que cuando una máquina pasa de servidor a controlador de dominio y luego

vuelve al estado de servidor pierde el acceso a un grupo mqm local. El síntoma es un error que indica que falta un grupo mqm local, por ejemplo:

```
>crtmqm qm0
AMQ8066:Local mqm group not found.
```

Para solucionar este problema, vuelva a crear el grupo mqm local utilizando las herramientas de gestión estándares de Windows. Puesto que toda la información de pertenencia a grupos se pierde, debe volver a incluir los usuarios de IBM MQ con privilegios en el grupo mqm local que acaba de crear. Si la máquina es un miembro del dominio, también debe añadir el grupo mqm de dominio al grupo mqm local, para otorgar a los ID de usuario IBM MQ de dominio con privilegios el nivel de autorización necesario.

Windows *Restricciones en los grupos anidados en Windows*

Existen restricciones en el uso de grupos anidados. Estas restricciones se deben en parte al nivel funcional del dominio y en parte a restricciones de IBM MQ.

Active Directory puede dar soporte a distintos tipos de grupos en un contexto de dominio dependiendo del nivel funcional del dominio. De forma predeterminada, los dominios de Windows 2003 están en el directorio " Windows 2000 mixed " nivel funcional. (Windows Server 2008 y Windows Server 2012 siguen el modelo de dominio Windows 2003 .) El nivel funcional del dominio determina los tipos de grupos soportados y el nivel de anidamiento permitido al configurar los ID de usuario en un entorno de dominio. Consulte la documentación de Active Directory para obtener información detallada sobre el Ámbito de grupo y los criterios de inclusión.

Además de los requisitos de Active Directory, se imponen restricciones adicionales para los ID utilizados por IBM MQ. Las API de red que utiliza IBM MQ no dan soporte a todas las configuraciones a las que da soporte el nivel funcional del dominio. Como resultado, IBM MQ no puede consultar la pertenencia a grupos de cualquier ID de dominio presente en un grupo Local de dominio que luego se anida en un grupo local. Además, no se da soporte al anidamiento múltiple de grupos globales y universales. No obstante, los grupos globales o universales anidados inmediatamente están soportados.

Windows *Autorización de usuarios para utilizar IBM MQ de forma remota*

Si tiene que crear e iniciar gestores de colas cuando esté conectado a IBM MQ de forma remota, debe tener el acceso de usuario `Crear objetos globales`.

Acerca de esta tarea

Nota: Los administradores tienen el acceso de usuario `Crear objetos globales` de forma predeterminada, así pues si usted es un administrador, puede crear e iniciar los gestores de colas cuando están conectados de forma remota sin alterar los derechos de usuario.

Si se está conectando a una máquina de Windows utilizando Terminal Services o una conexión de escritorio remoto y tiene problemas para crear, iniciar o suprimir un gestor de colas, esto puede ser debido a que no tiene el acceso de usuario `Crear objetos globales`.

El acceso de usuario `Crear objetos globales` limita a los usuarios autorizados a crear objetos en el espacio de nombres globales. Para que una aplicación pueda crear un objeto global, se debe ejecutar en el espacio de nombres global o el usuario en el que se está ejecutando la aplicación debe disponer del acceso de usuario `Crear objetos globales`.

Cuando se conecta de forma remota a una máquina Windows utilizando Terminal Services o una conexión de escritorio remoto, las aplicaciones se ejecutan en su propio espacio de nombres local. Si intenta crear o suprimir un gestor de colas utilizando IBM MQ Explorer o el mandato **crtmqm** o **dltmqm**, o iniciar un gestor de colas utilizando el mandato **strmqm**, se genera un error de autorización. Así se crea un FDC de IBM MQ con el ID de analizador XY132002.

El inicio de un gestor de colas utilizando IBM MQ Explorer o utilizando el mandato **amqmdain qmgr start** funciona correctamente porque estos mandatos no inician directamente el gestor de colas. En cambio, los mandatos envían la solicitud para iniciar el gestor de colas a un proceso independiente que se está ejecutando en el espacio de nombres global.

Si los distintos métodos de administración de IBM MQ no funcionan cuando se utiliza Terminal Services, intente establecer el derecho de usuario `Crear objetos globales`.

Procedimiento

1. Abra el panel Herramientas administrativas:

   Windows Server 2008 y Windows Server 2012
   Acceda a este panel utilizando **Panel de control** > **Sistema y mantenimiento** > **Herramientas administrativas**.

   Windows 8.1
   Acceda a este panel utilizando **Herramientas administrativas** > **Administración de equipos**
2. Efectúe una doble pulsación en **Directiva de seguridad local**.
3. Expanda **Directivas locales**.
4. Pulse **Asignación de derechos de usuario**.
5. Añada el usuario o grupo nuevo a la directiva `Crear objetos globales`.

Windows *El programa de salida de canal SSPI en Windows*

IBM MQ for Windows proporciona un programa de salida de seguridad, que se puede utilizar tanto en canales de mensajes como en canales MQI. La salida se suministra como código fuente y código objeto, y proporciona autenticación unidireccional y bidireccional.

La salida de seguridad utiliza la Interfaz del proveedor de soporte para seguridad (SSPI), que proporciona los recursos de seguridad integrados de las plataformas Windows.

La salida de seguridad proporciona los siguientes servicios de identificación y autenticación:

Autenticación unidireccional
Esto utiliza el soporte de autenticación de Windows NT LAN Manager (NTLM). NTLM permite a los servidores autenticar sus clientes. No permite que un cliente autentique un servidor, ni que un servidor autentique otro. NTLM se ha diseñado para un entorno de red en el que se da por supuesto que los servidores son genuinos. NTLM está soportado en todas las plataformas Windows soportadas en IBM WebSphere MQ 7.0.

Este servicio se suele utilizar en un canal MQI para permitir que un gestor de colas del servidor autentique una aplicación IBM MQ MQI client. Una aplicación cliente se identifica mediante el ID de usuario asociado con el proceso que se está ejecutando.

Para llevar a cabo la autenticación, la salida de seguridad en el extremo cliente de un canal adquiere una señal de autenticación de NTLM y envía la señal en un mensaje de seguridad a su asociado en el otro extremo del canal. La salida de seguridad del asociado pasa la señal a NTLM, el cual comprueba que la señal es auténtica. Si la salida de seguridad del asociado no está satisfecha con la autenticidad de la señal, indica al MCA que cierre el canal.

Autenticación bidireccional o mutua
Utiliza los servicios de autenticación de Kerberos. El protocolo Kerberos no da por supuesto que los servidores de un entorno de red son genuinos. Los servidores pueden autenticar clientes y otros servidores, y los clientes pueden autenticar servidores. Kerberos está soportado en todas las plataformas Windows soportadas en IBM WebSphere MQ 7.0.

Este servicio se puede utilizar en canales de mensajes y MQI. En un canal de mensajes, proporciona autenticación mutua de los dos gestores de colas. En un canal MQI, permite que el gestor de colas del servidor y la aplicación IBM MQ MQI client se autentiquen entre sí. Un gestor de colas se identifica por su nombre con el prefijo de la serie `ibmMQSeries/`. Una aplicación cliente se identifica mediante el ID de usuario asociado con el proceso que se está ejecutando.

Para realizar la autenticación mutua, la salida de seguridad inicial adquiere una señal de autenticación del servidor de seguridad Kerberos y envía la señal en un mensaje de seguridad a su asociado. La salida de seguridad del asociado pasa la señal al servidor de seguridad Kerberos, el cual comprueba que es auténtica. El servidor de seguridad Kerberos genera una segunda señal, que el asociado envía

en un mensaje de seguridad a la salida de seguridad inicial. La salida de seguridad inicial solicita al servidor Kerberos que compruebe que la segunda señal es auténtica. Durante este intercambio, si alguna de las salidas de seguridad no está satisfecha con la autenticidad de la señal enviada por la otra, indica al MCA que cierre el canal.

La salida de seguridad se suministra en formato fuente y objeto. Puede utilizar el código fuente como punto de partida para escribir sus propios programas de salida de canal o puede utilizar el módulo objeto tal como se suministra. El módulo objeto tiene dos puntos de entrada, uno para la autenticación unidireccional mediante el soporte para autenticación NTLM y el otro para la autenticación bidireccional mediante servicios de autenticación de Kerberos.

Para obtener más información sobre cómo funciona el programa de salida de canal SSPI y para ver instrucciones sobre cómo implementarlo, consulte Utilización de la salida de seguridad SSPI en sistemas Windows.

Windows *Aplicación de archivos de plantilla de seguridad en Windows*

La aplicación de una plantilla puede afectar a los valores de seguridad aplicados a los archivos y directorios de IBM MQ. Si utiliza la plantilla de alta seguridad, aplíquela antes de instalar IBM MQ.

Windows soporta archivos de plantilla de seguridad basados en texto que puede utilizar para aplicar valores de seguridad uniformes a uno o más sistemas con el complemento Configuración y análisis de seguridad de MMC. En particular, Windows proporciona varias plantillas que incluyen un rango de valores de seguridad con objeto de proporcionar niveles de seguridad específicos. Estas plantillas de seguridad predefinidas incluyen las plantillas Compatible, Segura y De alta seguridad.

La aplicación de una de estas plantillas puede afectar a los valores de seguridad aplicados a los archivos y directorios de IBM MQ. Si desea utilizar la plantilla De alta seguridad, configure la máquina antes de instalar IBM MQ.

Si aplica la plantilla de alta seguridad en una máquina en la que IBM MQ ya está instalado, todos los permisos que haya establecido en los archivos y directorios de IBM MQ se eliminarán. Puesto que estos permisos se eliminan, perderá el acceso al grupo *Administradores*, *mqm*, y, si procede, el acceso al grupo *Todos* desde los directorios de error.

Windows *Configuración de autorización adicional para aplicaciones Windows que se conectan a IBM MQ*

Es posible que la cuenta con la que se ejecutan los procesos de IBM MQ requiera autorización adicional antes de que se pueda otorgar acceso SYNCHRONIZE a los procesos de aplicaciones.

Acerca de esta tarea

Es posible que experimente problemas si tiene aplicaciones Windows, por ejemplo páginas ASP, que se conectan a IBM MQ y que están configuradas para ejecutarse a un nivel de seguridad superior al habitual.

IBM MQ requiere acceso SYNCHRONIZE para los procesos de aplicaciones a fin de coordinar ciertas acciones. Cuando una aplicación de servidor intenta por primera vez conectarse a un gestor de colas de IBM MQ modificará el acceso para otorgar autorización SYNCHRONIZE para administradores de IBM MQ. Sin embargo, es posible que la cuenta bajo la que se ejecutan los procesos de IBM MQ necesite autorización adicional antes de que se pueda otorgar el acceso solicitado.

Para configurar autorización adicional para el ID de usuario bajo el que se ejecutan los procesos de IBM MQ, realice los pasos siguientes:

Procedimiento

1. Inicie la herramienta Directiva de seguridad local, pulse **Configuración de seguridad**->**Directivas locales**->**Asignación de derechos de usuario** y pulse **Depurar programas**.
2. Efectúe una doble pulsación en **Depurar programas** y, a continuación, añada el ID de usuario de IBM MQ a la lista

Si el sistema está en un dominio Windows y el valor de directiva efectivo no está definido todavía, aunque el valor de directiva local esté definido, el ID de usuario debe autorizarse de la misma manera a nivel de dominio, utilizando la herramienta Directiva de seguridad de dominio.

IBM i Configuración de la seguridad en IBM i

La seguridad para IBM i se implementa utilizando el Gestor de autorizaciones sobre objetos (OAM) de IBM MQ y la seguridad a nivel de objeto de IBM i.

Consideraciones sobre seguridad que deben tenerse presentes al determinar la autorización de acceso a los objetos de IBM MQ.

Debe tener en cuenta las siguientes cuestiones cuando vaya a configurar las autorizaciones de los usuarios de la empresa:

1. Otorgue y revoque autorizaciones para los mandatos de IBM MQ for IBM i mediante los mandatos IBM i GRTOBJAUT y RVKOBJAUT.

   En la biblioteca QMQM, ciertos objetos no de mandato (*cmd) están definidos para tener la autorización ***PUBLIC** establecida en ***USE**. No cambie las autorizaciones de estos objetos ni utilice una lista de autorizaciones para otorgar autorización. Una autorización incorrecta puede comprometer la funcionalidad de IBM MQ.
2. Durante la instalación de IBM MQ for IBM i, se crean los siguientes perfiles de usuario especiales:

   QMQM
   Se utiliza principalmente para funciones internas sólo del producto. No obstante, se puede utilizar para ejecutar aplicaciones de confianza con MQCNO_FASTPATH_BINDINGS. Consulte Conectarse a un gestor de colas mediante la llamada MQCONNX.

   QMQMADM
   Se utiliza como perfil de grupo para los administradores de IBM MQ. El perfil de grupo otorga acceso a los mandatos CL y a los recursos de IBM MQ.

   Si se utiliza SBMJOB para someter programas que llaman a mandatos IBM MQ, USER no debe establecerse explícitamente en QMQMADM. En su lugar, establezca USER en QMQM u otro perfil de usuario que tenga QMQMADM especificado como grupo.
3. Si va a enviar mandatos de canal a gestores de colas remotos, asegúrese de que su perfil de usuario es miembro del grupo QMQMADM en el sistema de destino. Para obtener una lista de los mandatos de canal PCF y MQSC, consulte Mandatos CL de IBM MQ for IBM i.
4. El conjunto de grupos asociado a un usuario se almacena en memoria caché cuando el OAM calcula las autorizaciones de grupo.

   Todos los cambios realizados en la pertenencia a grupos de un usuario después de que el conjunto de grupos se ha almacenado en la memoria caché no se reconocerán hasta que se reinicie el gestor de colas o se ejecute RFRMQMAUT para renovar la seguridad.
5. Limite el número de usuarios que poseen autorización para trabajar con los mandatos que sean especialmente delicados. Entre ellos se encuentran los siguientes mandatos:
   - Creación de un gestor de colas de mensajes (CRTMQM)
   - Borrado de un gestor de colas de mensajes (DLTMQM)
   - Inicio de un gestor de colas de mensajes (STRMQM)
   - Terminación de un gestor de colas de mensajes (ENDMQM)
   - Inicio de un servidor de mandatos (STRMQMCSVR)
   - Terminación de un servidor de mandatos (ENDMQMCSVR)
6. Las definiciones de canal contienen una especificación de programa de salida de seguridad. Requieren consideraciones especiales la creación y la modificación de canales. Encontrará información detallada sobre las salidas de seguridad en “Visión general de las salidas de seguridad” en la página 107.
7. Se pueden sustituir los programas de salida de canal y de supervisor desencadenante. Ha de ser el programador quien se encargue de la seguridad de estas sustituciones.

IBM i Gestor de autorizaciones sobre objetos (OAM) en IBM i

El gestor de autorizaciones sobre objetos (OAM) gestiona las autorizaciones de los usuarios para manipular objetos de IBM MQ, incluyendo colas y definiciones de proceso. También proporciona una interfaz de mandatos mediante la cual puede otorgar o revocar la autorización de acceso a un objeto para un grupo de usuarios específico. La decisión de permitir el acceso a un recurso la toma el OAM, y el gestor de colas actúa según la decisión tomada. Si el OAM no puede tomar una decisión, el gestor de colas impide el acceso a dicho recurso.

Mediante el OAM, puede controlar:

- El acceso a objetos de IBM MQ mediante la interfaz de cola de mensajes (MQI). Cuando un programa de aplicación intenta acceder a un objeto, el OAM comprueba que el perfil de usuario que realiza la solicitud posee autorización para la operación solicitada.

  En concreto, esto significa que las colas y los mensajes de las colas se pueden proteger contra accesos no autorizados.
- El permiso para utilizar mandatos PCF y MQSC.

Es posible que distintos grupos de usuarios tengan diferentes autorizaciones de acceso al mismo objeto. Por ejemplo, para una cola específica, un grupo podría realizar las operaciones de transferir y obtener; otro grupo podría tener autorización únicamente para examinar la cola (MQGET con la opción de examinar). De forma parecida, algunos grupos podrían tener autorización para transferir y obtener en una cola, pero no tenerla para modificar o suprimir la cola.

Mandatos IBM MQ for IBM i y realizar operaciones en objetos de IBM MQ for IBM i

IBM i Autorizaciones de IBM MQ en IBM i

Para acceder a objetos IBM MQ, necesita autorización para emitir el mandato y para acceder al objeto referenciado. Los administradores tienen acceso a todos los recursos de IBM MQ.

El acceso a los objetos de IBM MQ se controla mediante las autorizaciones para:

1. Emitir el mandato IBM MQ
2. Acceder a los objetos de IBM MQ a los que hace referencia el mandato

Todos los mandatos CL de IBM MQ for IBM i se suministran con un propietario de QMQM, y el perfil de administración (QMQMADM) tiene derechos *USE con el acceso *PUBLIC establecido en *EXCLUDE.

Nota: El programa instalador con licencia de IBM MQ para IBM i utiliza el programa QSRDUPER para duplicar objetos de mandato (*CMD) en QSYS. En IBM i V5R4 y posterior, el programa QSRDUPER se modificó de forma que el comportamiento predeterminado sea crear un mandato proxy en lugar de una duplicación de un mandato original. Un mandato proxy redirige la ejecución del mandato a otro mandato y tiene un atributo PRX. Si existe un mandato proxy en la biblioteca QSYS con el mismo nombre que el mandato que se está copiando, las autorizaciones privadas al mandato proxy no se otorgan al mandato en la biblioteca del producto. Los intentos de solicitar o ejecutar el mandato proxy en QSYS comprueban la autorización del mandato de destino en la biblioteca del producto. Todos los cambios en la autorización para objetos *CMD deben, por consiguiente, realizarse en la biblioteca del producto (QMQM) y los de QSYS no necesitan modificarse. Por ejemplo:

```
GRTOBJAUT OBJ(QMQM/DSPMQMQ) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
```

Los cambios en la estructura de autorización de algunos de los mandatos CL del producto permiten el uso púbico de estos mandatos, si tiene la autorización del OAM necesaria sobre los objetos IBM MQ para realizar estos cambios.

Para ser un administrador de IBM MQ en IBM i, debe ser miembro del *grupo QMQMADM*. Este grupo tiene propiedades similares a las del grupo mqm en los sistemas UNIX, Linux yWindows. En particular, el grupo QMQMADM se crea al instalar IBM MQ for IBM i y los miembros del grupo QMQMADM tienen acceso a todos los recursos de IBM MQ en el sistema. También tiene acceso a todos los recursos IBM MQ si tiene autorización *ALLOBJ.

Los administradores pueden utilizar mandatos CL para administrar IBM MQ. Uno de estos mandatos es GRTMQMAUT, que se utiliza para conceder autorizaciones a otros usuarios. Otro mandato, STRMQMMQSC, permite que un administrador emita mandatos MQSC a un gestor de colas local.

Conceptos relacionados

"Autorización para administrar IBM MQ en IBM i" en la página 85

IBM i *Autorizaciones de acceso para los objetos de IBM MQ en IBM i*

Autorizaciones de acceso necesarias para ejecutar mandatos CL de IBM MQ.

IBM MQ for IBM i categoriza los mandatos CL del producto en dos grupos:

Grupo 1

Los usuarios deben estar en el grupo de usuarios QMQMADM, o bien tener la autorización *ALLOBJ para procesar estos mandatos. Los usuarios que tienen una de estas autorizaciones pueden procesar todos los mandatos de todas las categorías y no necesitan ninguna otra autorización.

Nota: Estas autorizaciones alteran temporalmente cualquier autorización del OAM.

Estos mandatos se pueden agrupar como se indica a continuación:

- Mandatos de servidor de mandatos
  - ENDMQMCSVR, Finalizar el servidor de mandatos de IBM MQ
  - STRMQMCSVR, Iniciar el servidor de mandatos de IBM MQ
- Mandato de manejador de la cola de mensajes no entregados
  - STRMQMDLQ, Iniciar manejador de la cola de mensajes no entregados de IBM MQ
- Mandato de escucha
  - ENDMQMLSR, Finalizar escucha de IBM MQ
  - STRMQMLSR, Iniciar escucha no de objeto
- Mandatos de recuperación desde soporte
  - RCDMQMIMG, Registrar imagen de objeto de IBM MQ
  - RCRMQMOBJ, Volver a crear objeto de IBM MQ
  - WRKMQMTRN, Trabajar con transacciones de IBM MQ MQ
- Mandatos de gestor de colas
  - CRTMQM, Crear gestor de colas de mensajes
  - DLTMQM, Suprimir gestor de colas de mensajes
  - ENDMQM, Finalizar gestor de colas de mensajes
  - STRMQM, Iniciar gestor de colas de mensajes
- Mandatos de seguridad
  - GRTMQMAUT, Otorgar autorización sobre objeto de IBM MQ
  - RVKMQMAUT, Revocar autorización sobre objeto de IBM MQ
- Mandato de rastreo
  - TRCMQM, Rastrear trabajo de IBM MQ
- Mandatos de transacción
  - RSVMQMTRN, Resolver transacción de IBM MQ
- Mandatos de supervisor desencadenante
  - STRMQMTRM, Iniciar supervisor desencadenante
- Mandatos IBM MQSC
  - RUNMQSC, Ejecutar mandatos IBM MQSC

– STRMQMMQSC, Iniciar mandatos IBM MQSC

Grupo 2

El resto de los mandatos, para los que se requiere dos niveles de autorización:

1. Autorización de IBM i para ejecutar el mandato. Un administrador de IBM MQ establece esto mediante el mandato **GRTOBJAUT** para alterar temporalmente la restricción *PUBLIC(*EXCLUDE) para un usuario o grupo de usuarios.

   Por ejemplo:

   ```
   GRTOBJAUT OBJ(QMQM/DSPMQMQ) OBJTYPE(*CMD) USER(MQUSER) AUT(*USE)
   ```

2. Autorización de IBM MQ para manipular los objetos de IBM MQ asociados con el mandato o los mandatos, dada la autorización correcta de IBM i en el paso 1.

   Esta autorización está controlada por el usuario que tiene la autorización adecuada del OAM para la acción necesaria, establecida por un administrador de IBM MQ mediante el mandato **GRTMQMAUT**

   Por ejemplo:

   ```
   GRTMQMAUT *connect authority to the queue manager + *admchg authority to
           the queue
   ```

Los mandatos se pueden agrupar como se indica a continuación:

- Mandatos de canal
  - CHGMQMCHL, Cambiar el canal de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *admchg sobre el canal.
  - CPYMQMCHL, Copiar el canal de IBM MQ

    Requiere la autorización *connect y *admcrt sobre el gestor de colas, la autorización *admdsp sobre el tipo de canal predeterminado que se va a copiar y la autorización *admcrt sobre la clase de objeto del canal.

    Por ejemplo, para copiar un canal emisor, se necesita autorización *admdsp sobre el canal SYSTEM.DEF.SENDER
  - CRTMQMCHL, Crear el canal de IBM MQ

    Requiere la autorización *connect y *admcrt sobre el gestor de colas, la autorización *admdsp sobre el tipo de canal predeterminado que se va a crear y la autorización *admcrt sobre la clase de objeto del canal.

    Por ejemplo, para crear un canal emisor, se necesita autorización *admdsp sobre el canal SYSTEM.DEF.SENDER
  - DLTMQMCHL, Suprimir canal de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *admdlt sobre el canal.
  - RSVMQMCHL, Resolver el canal de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *ctrlx sobre el canal.
- Mandatos de Visualizar

  Para procesar los mandatos DSP, debe otorgar al usuario las autorizaciones `*connect` y `*admdsp` sobre el gestor de colas, junto con alguna de las opciones concretas que aparecen en la lista siguiente:
  - DSPMQM, Visualizar gestor de colas de mensajes

- DSPMQMAUT, Visualizar autorización sobre objeto de IBM MQ
- DSPMQMAUTI, Visualizar información de autenticación de IBM MQ - *admdsp en el objeto de información de autenticación
- DSPMQMCHL, Visualizar el canal de IBM MQ - *admdsp sobre el canal
- DSPMQMCSVR, Visualizar el servidor de mandatos de IBM MQ
- DSPMQMNL, Visualizar lista de nombres de IBM MQ - *admdsp sobre la lista de nombres
- DSPMQMOBJN, Visualizar nombres de objeto de IBM MQ
- DSPMQMPRC, Visualizar proceso de IBM MQ - *admdsp sobre el proceso
- DSPMQMQ, Visualizar cola de IBM MQ - *admdsp sobre la cola
- DSPMQMTOP, Visualizar tema de IBM MQ - *admdsp sobre el tema

- Mandatos de Trabajar con

  Para procesar los mandatos WRK y visualizar el panel de opciones, debe otorgar al usuario las autorizaciones *connect y *admdsp sobre el gestor de colas, junto con alguna de las opciones concretas que aparecen en la lista siguiente:

  - WRKMQM, Trabajar con gestores de colas de mensajes
  - WRKMQMAUT, Trabajar con autorización sobre objeto de IBM MQ
  - WRKMQMAUTD, Trabajar con datos de autorización sobre objeto de IBM MQ
  - WRKMQMAUTI, Trabajar con información de autenticación de IBM MQ
    - *admchg para el mandato Cambiar objeto de información de autenticación de IBM MQ
    - *admcrt para el mandato Crear y copiar objeto de información de autenticación de IBM MQ.
    - *admdlt para el mandato Suprimir objeto de información de autenticación de IBM MQ.
    - *admdsp mandato Visualizar objeto de información de autenticación de IBM MQ.
  - WRKMQMCHL, Trabajar con canal de IBM MQ

    Requiere las siguientes autorizaciones:

    - *admchg para el mandato Cambiar canal de IBM MQ.
    - *admclr para el mandato Borrar canal de IBM MQ.
    - *admcrt para el mandato Crear y copiar canal de IBM MQ.
    - *admdlt para el mandato Suprimir canal de IBM MQ.
    - *admdsp para el mandato Visualizar canal de IBM MQ.
    - *ctrl para el mandato Iniciar canal de IBM MQ.
    - *ctrl para el mandato Finalizar canal de IBM MQ.
    - *ctrl para el mandato Sondear canal de IBM MQ.
    - *ctrlx para el mandato Restablecer canal de IBM MQ.
    - *ctrlx para el mandato Resolver canal de IBM MQ.
  - WRKMQMCHST, Trabajar con estado de canal de IBM MQ

    Requiere la autorización *admdsp sobre el canal.
  - WRKMQMCL, Trabajar con clústeres de IBM MQ
  - WRKMQMCLQ, Trabajar con colas de clúster de IBM MQ
  - WRKMQMCLQM, Trabajar con el gestor de colas de clúster de IBM MQ
  - WRKMQMLSR, Trabajar con escucha de IBM MQ
  - WRKMQMMSG, Trabajar con mensajes de IBM MQ

    Requiere la autorización *browse sobre la cola
  - WRKMQMNL, Trabajar con listas de nombres de IBM MQ

Requiere las siguientes autorizaciones:

- *admchg para el mandato Cambiar lista de nombres de IBM MQ.
- *admcrt para el mandato Crear y copiar lista de nombres de IBM MQ.
- *admdlt para el mandato Suprimir lista de nombres de IBM MQ.
- *admdsp para el mandato Visualizar lista de nombres de IBM MQ.

– WRKMQMPRC, Trabajar con procesos de IBM MQ

Requiere las siguientes autorizaciones:

- *admchg para el mandato Cambiar proceso de IBM MQ.
- *admcrt para el mandato Crear y copiar proceso de IBM MQ.
- *admdlt para el mandato Suprimir proceso de IBM MQ.
- *admdsp para el mandato Visualizar proceso de IBM MQ.

– WRKMQMQ, Trabajar con colas de IBM MQ

Requiere las siguientes autorizaciones:

- *admchg para el mandato Cambiar cola de IBM MQ.
- *admclr para el mandato Borrar cola de IBM MQ.
- *admcrt para el mandato Crear y copiar cola de IBM MQ.
- *admdlt para el mandato Suprimir cola de IBM MQ.
- *admdsp para el mandato Visualizar cola de IBM MQ.

– WRKMQMQSTS, Trabajar con estado de cola de IBM MQ

– WRKMQMTOP, Trabajar con temas de IBM MQ

Requiere las siguientes autorizaciones

- *admchg para el mandato Cambiar tema de IBM MQ.
- *admcrt para el mandato Crear y copiar tema de IBM MQ.
- *admdlt para el mandato Suprimir tema de IBM MQ.
- *admdsp para el mandato Visualizar tema de IBM MQ.

– WRKMQMSUB, Trabajar con suscripciones de IBM MQ

• Otros mandatos de canal

Para procesar los mandatos de canal, debe otorgar al usuario las autorizaciones específicas que aparecen en la lista siguiente:

– ENDMQMCHL, Finalizar canal de IBM MQ

Requiere la autorización *connect sobre el gestor de colas y la autorización *allmqi sobre la cola de transmisión asociada al canal.

– ENDMQMLSR, Finalizar escucha de IBM MQ

Esto requiere la autorización *connect sobre el gestor de colas y la autorización *ctrl sobre el objeto de escucha especificado.

– PNGMQMCHL, Hacer ping en el canal de IBM MQ

Esto requiere autorización *connect y *inq al gestor de objetos y autorización *ctrl al objeto de canal.

– RSTMQMCHL, Restablecer canal de IBM MQ

Requiere la autorización *connect sobre el gestor de colas.

– STRMQMCHL, Iniciar canal de IBM MQ

Esto requiere autorización *connect para el gestor de colas y autorización *ctrl para el objeto de canal.

- STRMQMCHLI, Iniciar iniciador de canal de IBM MQ

  Requiere las autorizaciones *connect e *inq sobre el gestor de colas, y la autorización *allmqi sobre la cola de iniciación asociada a la cola de transmisión del canal.

- STRMQMLSR, Iniciar escucha de IBM MQ

  Requiere la autorización *connect sobre el gestor de colas y la autorización *ctrl sobre el objeto de escucha especificado.

- Otros mandatos:

  Para procesar los mandatos siguientes, debe otorgar al usuario las autorizaciones específicas que aparecen en esta lista:

  - CCTMQM, Conectar a gestor de colas de mensajes

    Esto no necesita autoridad sobre objeto de IBM MQ.

  - CHGMQM, Cambiar gestor de colas de mensajes

    Esto requiere autorización *connect y *admchg para el gestor de colas.

  - CHGMQMAUTI, Cambiar información de autenticación de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *admchg y *admdsp sobre el objeto de información de autenticación.

  - CHGMQMNL, Cambiar lista de nombres de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y *admchg sobre la lista de nombres.

  - CHGMQMPRC, Cambiar proceso de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y *admchg sobre el proceso.

  - CHGMQMQ, Cambiar cola de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y *admchg sobre la cola.

  - CLRMQMQ, Borrar cola de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y *admclr sobre la cola.

  - CPYMQMAUTI, Copiar información de autenticación de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *admdsp sobre el objeto de información de autenticación y la autorización *admcrt sobre la clase de objeto de información de autenticación.

  - CPYMQMNL, Copiar lista de nombres de IBM MQ

    Esto requiere autorización *connect y *admcrt para el gestor de colas.

  - CPYMQMPRC, Copiar proceso de IBM MQ

    Esto requiere autorización *connect y *admcrt para el gestor de colas.

  - CPYMQMQ, Copiar cola de IBM MQ

    Esto requiere autorización *connect y *admcrt para el gestor de colas.

  - CRTMQMAUTI, Crear información de autenticación de IBM MQ

    Requiere la autorización *connect sobre el gestor de colas y la autorización *admdsp sobre el objeto de información de autenticación y la autorización *admcrt sobre la clase de objeto de información de autenticación.

  - CRTMQMNL, Crear lista de nombres de IBM MQ

    Requiere la autorización *connect y *admcrt sobre el gestor de colas y la autorización *admdsp sobre la lista de nombres predeterminada.

  - CRTMQMPRC, Crear proceso de IBM MQ

    Requiere la autorización *connect y *admcrt sobre el gestor de colas y *admdsp sobre el proceso predeterminado.

- CRTMQMQ, Crear cola de IBM MQ

  Requiere la autorización `*connect` y `*admcrt` sobre el gestor de colas y `*admdsp` sobre la cola predeterminada.
- CVTMQMDTA, mandato Convertir tipo de datos de IBM MQ

  Esto no necesita autoridad sobre objeto de IBM MQ.
- DLTMQMAUTI, Suprimir información de autenticación de IBM MQ

  Esto requiere autorización `*connect` para el gestor de colas y autorización `*ctrlx` para el objeto de información de autenticación.
- DLTMQMNL, Suprimir lista de nombres de IBM MQ

  Requiere la autorización `*connect` sobre el gestor de colas y `*admdlt` sobre la lista de nombres.
- DLTMQMPRC, Suprimir proceso de IBM MQ

  Requiere la autorización `*connect` sobre el gestor de colas y `*admdlt` sobre el proceso.
- DLTMQMQ, Suprimir cola de IBM MQ

  Requiere la autorización `*connect` sobre el gestor de colas y `*admdlt` sobre la cola.
- DSCMQM, Desconectar de gestor de colas de mensajes

  Esto no necesita autoridad sobre objeto de IBM MQ.
- RFRMQMAUT, Renovar seguridad

  Requiere la autorización `*connect` sobre el gestor de colas.
- RFRMQMCL, Renovar clúster

  Requiere la autorización `*connect` sobre el gestor de colas.
- RSMMQMCLQM, Reanudar gestor de colas de clúster

  Requiere la autorización `*connect` sobre el gestor de colas.
- RSTMQMCL, Restablecer clúster

  Requiere la autorización `*connect` sobre el gestor de colas.
- SPDMQMCLQM, Suspender gestor de colas de clúster

  Requiere la autorización `*connect` sobre el gestor de colas.

IBM i *Autorizaciones de acceso en IBM i*

Lea esta información para entender los mandatos de autorización de acceso.

Las autorizaciones definidas mediante la palabra clave AUT en los mandatos GRTMQMAUT y RVKMQMAUT se pueden clasificar de la siguiente manera:

- Autorizaciones relacionadas con las llamadas MQI
- Autorizaciones relacionadas con los mandatos de administración
- Autorizaciones de contexto
- Autorizaciones generales, es decir, para llamadas MQI, para mandatos o para ambos

Las tablas que figuran más abajo muestran las distintas autorizaciones que utilizan el parámetro AUT para llamadas MQI, llamadas de contexto, mandatos MQSC y PCF, y operaciones genéricas.

Tabla 15. Autorizaciones para llamadas MQI

AUT	Descripción
*ALTUSR	Permitir que se utilice la autorización de otro usuario para llamadas MQOPEN y MQPUT1.

Tabla 15. Autorizaciones para llamadas MQI (continuación)	
AUT	**Descripción**
*BROWSE	Recuperar un mensaje de una cola emitiendo una llamada MQGET con la opción BROWSE.
*CONNECT	Conectar la aplicación con el gestor de colas especificado emitiendo una llamada MQCONN.
*GET	Recuperar un mensaje de una cola emitiendo una llamada MQGET.
*INQ	Efectuar una consulta sobre una cola específica emitiendo una llamada MQINQ.
*PUB	Abrir un tema para publicar un mensaje utilizando una llamada MQPUT.
*PUT	Transferir un mensaje a una cola específica emitiendo una llamada MQPUT.
*RESUME	Reanudar una suscripción utilizando una llamada MQSUB.
*SET	Establecer los atributos de una cola de la MQI emitiendo una llamada MQSET. Si abre una cola para varias opciones, debe tener autorización sobre todas ellas.
*SUB	Crear, modificar o reanudar una suscripción en un tema utilizando una llamada MQSUB.

Tabla 16. Autorizaciones para llamadas de contexto	
AUT	**Descripción**
*PASSALL	Pasar todo el contexto de la cola especificada. Todos los campos de contexto se copian de la solicitud original.
*PASSID	Pasar el contexto de identidad en la cola especificada. El contexto de identidad es el mismo que el de la solicitud.
*SETALL	Establecer todo el contexto de la cola especificada. Esta autorización la utilizan programas de utilidad especiales del sistema.
*SETID	Establecer el contexto de identidad de la cola especificada. Esta autorización la utilizan programas de utilidad especiales del sistema.

Tabla 17. Autorizaciones para llamadas MQSC y PCF	
AUT	**Descripción**
*ADMCHG	Cambiar los atributos del objeto especificado.
*ADMCLR	Vaciar el objeto especificado (sólo el mandato PCF Vaciar objeto).
*ADMCRT	Crear objetos del tipo especificado.
*ADMDLT	Suprimir el objeto especificado.
*ADMDSP	Visualizar los atributos del objeto especificado.

Tabla 18. Autorizaciones para operaciones genéricas	
AUT	**Descripción**
*ALL	Utilizar todas las operaciones aplicables al objeto. La autorización `all` equivale a la unión de las autorizaciones `alladm`, `allmqi` y `system` adecuadas al tipo de objeto.
*ALLADM	Ejecutar todas las operaciones de administración aplicables al objeto.

Tabla 18. Autorizaciones para operaciones genéricas (continuación)	
AUT	**Descripción**
*ALLMQI	Utilizar todas las llamadas MQI aplicables al objeto.
*CTRL	Controlar el arranque y el cierre de canales, escuchas y servicios.
*CTRLX	Restablecer el número de secuencia y resolver canales pendientes

IBM i *Utilización de los mandatos de autorización de acceso en IBM i*

Lea esta información para obtener información sobre los mandatos de autorización de acceso, y utilice los ejemplos de mandatos.

Utilización del mandato GRTMQMAUT

Si posee la autorización necesaria, puede utilizar el mandato GRTMQMAUT para otorgar a un perfil de usuario o grupo de usuarios autorización para acceder a un determinado objeto. Los ejemplos que figuran a continuación ilustran cómo se utiliza el mandato GRTMQMAUT:

1. ```
   GRTMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
             AUT(*BROWSE *PUT) MQMNAME('saturn.queue.manager')
   ```

   En este ejemplo:

   - RED.LOCAL.QUEUE es el nombre del objeto.
   - *LCLQ (cola local) es el tipo de objeto.
   - GROUPA es el nombre de un perfil de usuario en el sistema para el que las autorizaciones se van a cambiar. Este perfil se puede utilizar como perfil de grupo para otros usuarios.
   - *BROWSE y *PUT son las autorizaciones que se van a otorgar sobre la cola especificada.

     *BROWSE añade autorización para examinar los mensajes de la cola (emitir MQGET con la opción de examinar).

     *PUT añade autorización para poner (MQPUT) mensajes en la cola.
   - saturn.queue.manager es el nombre del gestor de colas.
2. El siguiente mandato otorga a los usuarios JACK y JILL todas las autorizaciones aplicables, sobre todas las definiciones de proceso, del gestor de colas predeterminado.

   ```
   GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER(JACK JILL) AUT(*ALL)
   ```
3. El siguiente mandato otorga al usuario GEORGE autorización para poner un mensaje en la cola ORDERS, en el gestor de colas TRENT.

   ```
   GRTMQMAUT OBJ(TRENT) OBJTYPE(*MQM) USER(GEORGE) AUT(*CONNECT) MQMNAME (TRENT)
   GRTMQMAUT OBJ(ORDERS) OBJTYPE(*Q) USER(GEORGE) AUT(*PUT) MQMNAME (TRENT)
   ```

Utilización del mandato RVKMQMAUT

Si posee la autorización necesaria, puede utilizar el mandato RVKMQMAUT para eliminar del perfil de usuario o del grupo de usuarios una autorización ya otorgada para acceder a un determinado objeto. Los ejemplos que figuran a continuación ilustran cómo se utiliza el mandato RVKMQMAUT:

1. ```
   RVKMQMAUT OBJ(RED.LOCAL.QUEUE) OBJTYPE(*LCLQ) USER(GROUPA) +
   AUT(*PUT) MQMNAME('saturn.queue.manager')
   ```

Se quita del grupo GROUPA la autorización (otorgada en el ejemplo anterior) para poner mensajes en la cola especificada.

2.
```
RVKMQMAUT OBJ(PAY*) OBJTYPE(*Q) USER(*PUBLIC) AUT(*GET) +
MQMNAME(PAYROLLQM)
```

La autorización para obtener mensajes procedentes de cualquier cola con un nombre que empiece por los caracteres PAY, que pertenece al gestor de colas PAYROLLQM, se elimina de todos los usuarios del sistema, a menos que ellos o un grupo al que pertenezcan, se hayan autorizado por separado.

Utilización del mandato DSPMQMAUT

El mandato Visualizar autorización de MQM (DSPMQMAUT) muestra, para el objeto y usuario especificados, la lista de autorizaciones que el usuario posee sobre el objeto. El siguiente ejemplo ilustra cómo se utiliza este mandato:

```
DSPMQMAUT OBJ(ADMINNL) OBJTYPE(*NMLIST) USER(JOE) OUTPUT(*PRINT) +
MQMNAME(ADMINQM)
```

Utilización del mandato RFRMQMAUT

El mandato Renovar seguridad de MQM (RFRMQMAUT) le permite actualizar inmediatamente la información del grupo de autorizaciones del OAM, reflejando los cambios realizados a nivel de sistema operativo, sin necesidad de detener y reiniciar el gestor de colas. El siguiente ejemplo ilustra cómo se utiliza este mandato:

```
RFRMQMAUT MQMNAME(ADMINQM)
```

IBM i Tablas de especificación de autorizaciones en IBM i

Utilice esta información para determinar qué autorización es necesaria para utilizar llamadas API específicas, y opciones específicas de esas llamadas, en objetos de cola, objetos de proceso y objetos de gestor de colas.

Las tablas de especificación de autorizaciones en Tabla 19 en la página 167 definen de forma precisa cómo funcionan las autorizaciones y las restricciones que se aplican. Las tablas se aplican a estas situaciones:

- Aplicaciones que emiten llamadas MQI
- Programas de administración que emiten mandatos MQSC como mandatos PCF de escape
- Programas de administración que emiten mandatos PCF

En esta sección, la información se presenta como un conjunto de tablas que especifican los datos siguientes:

Acción que se va a realizar
Opción MQI, mandato MQSC o mandato PCF.

Objeto de control de acceso
Cola, definición de proceso, gestor de colas, lista de nombres, canal, canal de conexión de cliente, escucha, servicio u objeto de información de autenticación.

Autorización necesaria
Expresada como constante de tipo MQZAO_.

En las tablas, las constantes que tienen el prefijo MQZAO_ corresponden a las palabras clave en la lista de autorizaciones de los mandatos **GRTMQMAUT** y **RVKMQMAUT** de una determinada entidad. Por ejemplo, MQZAO_BROWSE corresponde con la palabra clave *BROWSE; asimismo, la palabra clave MQZAO_SET_ALL_CONTEXT corresponde con la palabra clave *SETALL y así sucesivamente. Estas constantes están definidas en el archivo de cabecera cmqzc.h que se proporciona con el producto.

Autorizaciones de MQI

Una aplicación puede emitir determinadas llamadas y opciones MQI sólo si el identificador de usuario bajo el que se está ejecutando (o cuyas autorizaciones puede asumir) tiene la autorización pertinente.

Hay cuatro llamadas MQI que pueden requerir comprobaciones de autorización: MQCONN, MQOPEN, MQPUT1 y MQCLOSE.

Para MQOPEN y MQPUT1, la comprobación de autorización se efectúa en el nombre del objeto que se está abriendo y no en el nombre o nombres resultantes de la resolución de nombre. Por ejemplo, una aplicación puede tener autorización para abrir una cola alias sin tener autorización para abrir la cola base en la que se resuelve la cola alias. La regla es que la comprobación se realiza en la primera definición encontrada durante el proceso de resolución de nombres que no es un alias de gestor de colas, a menos que la definición de alias de gestor de colas se abra directamente; es decir, su nombre aparece en el campo *ObjectName* del descriptor de objeto. La autoridad siempre es necesaria para el objeto concreto que se está abriendo; en algunos casos, es necesaria la autoridad adicional independiente de la cola obtenida a través de una autorización para el objeto del gestor de colas.

Tabla 19 en la página 167, Tabla 20 en la página 167, Tabla 21 en la página 168 y Tabla 22 en la página 168 resumen las autorizaciones necesarias para cada llamada.

Nota: En estas tablas no se mencionan listas de nombres, canales, canales de conexión de cliente, escuchas, servicios u objetos de información de autenticación. Esto se debe a que ninguna de las autorizaciones se aplica a estos objetos, salvo MQOO_INQUIRE, para la que se aplican las mismas autorizaciones que para los demás objetos.

Tabla 19. Autorización de seguridad necesaria para llamadas MQCONN

Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
Opción MQCONN	No aplicable	No aplicable	MQZAO_CONNECT

Tabla 20. Autorización de seguridad necesaria para llamadas MQOPEN

Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
MQOO_INQUIRE	MQZAO_INQUIRE **(“2” en la página 169)**	MQZAO_INQUIRE **(“2” en la página 169)**	MQZAO_INQUIRE **(“2” en la página 169)**
MQOO_BROWSE	MQZAO_BROWSE	No aplicable	No se comprueba
MQOO_INPUT_*	MQZAO_INPUT	No aplicable	No se comprueba
MQOO_SAVE_ ALL_CONTEXT **(“3” en la página 169)**	MQZAO_INPUT	No aplicable	No aplicable
MQOO_OUTPUT (Cola normal) **(“4” en la página 169)**	MQZAO_OUTPUT	No aplicable	No aplicable
MQOO_PASS_ IDENTITY_CONTEXT **(“5” en la página 169)**	MQZAO_PASS_ IDENTITY_CONTEXT	No aplicable	No se comprueba
MQOO_PASS_ALL_ CONTEXT **(“5” en la página 169, “6” en la página 169)**	MQZAO_PASS _ALL_CONTEXT	No aplicable	No se comprueba

Tabla 20. Autorización de seguridad necesaria para llamadas MQOPEN (continuación)			
Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
MQOO_SET_ IDENTITY_CONTEXT **(“5” en la página 169, “6” en la página 169)**	MQZAO_SET_ IDENTITY_CONTEXT	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“7” en la página 169)**
MQOO_SET_ ALL_CONTEXT **(“5” en la página 169, “8” en la página 169)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“7” en la página 169)**
MQOO_OUTPUT (cola de transmisión) **(“9” en la página 169)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“7” en la página 169)**
MQOO_SET	MQZAO_SET	No aplicable	No se comprueba
MQOO_ALTERNATE_ USER_AUTHORITY	**(“10” en la página 169)**	**(“10” en la página 169)**	MQZAO_ALTERNATE_ USER_AUTHORITY **(“10” en la página 169, “11” en la página 169)**

Tabla 21. Autorización de seguridad necesaria para llamadas MQPUT1			
Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
MQPMO_PASS_ IDENTITY_CONTEXT	MQZAO_PASS_ IDENTITY_CONTEXT **(“12” en la página 169)**	No aplicable	No se comprueba
MQPMO_PASS_ALL _CONTEXT	MQZAO_PASS_ ALL_CONTEXT **(“12” en la página 169)**	No aplicable	No se comprueba
MQPMO_SET_ IDENTITY_CONTEXT	MQZAO_SET_ IDENTITY_CONTEXT **(“12” en la página 169)**	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“7” en la página 169)**
MQPMO_SET_ ALL_CONTEXT	MQZAO_SET_ ALL_CONTEXT **(“12” en la página 169)**	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“7” en la página 169)**
(Cola de transmisión) **(“9” en la página 169)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“7” en la página 169)**
MQPMO_ALTERNATE_ USER_AUTHORITY	**(“13” en la página 169)**	No aplicable	MQZAO_ALTERNATE_ USER_AUTHORITY **(“11” en la página 169)**

Tabla 22. Autorización de seguridad necesaria para llamadas MQCLOSE			
Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCO_DELETE	MQZAO_DELETE **(“14” en la página 169)**	No aplicable	No aplicable

Tabla 22. Autorización de seguridad necesaria para llamadas MQCLOSE (continuación)			
Autorización necesaria para:	**Objeto de cola (“1” en la página 169)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCO_DELETE _PURGE	MQZAO_DELETE **(“14” en la página 169)**	No aplicable	No aplicable

Notas para las tablas:

1. Si se va abrir una cola modelo:
   - Para la cola modelo, es necesaria la autorización MQZAO_DISPLAY además de la autorización para abrir la cola modelo correspondiente al tipo de acceso para el que se está efectuando la apertura.
   - La autorización MQZAO_CREATE no es necesaria para crear la cola dinámica.
   - El identificador de usuario utilizado para abrir la cola modelo se otorga automáticamente a todas las autorizaciones específicas de la cola (equivalentes a MQZAO_ALL) para la cola dinámica creada.
2. Se comprueba el objeto de cola, proceso, lista de nombres o gestor de colas, dependiendo del tipo de objeto que vaya a abrirse.
3. También debe especificarse MQOO_INPUT_*. Esta opción es válida para una cola local, modelo o alias.
4. Esta comprobación se realiza en todos los casos de salida, excepto en el caso especificado en la nota “9” en la página 169.
5. También debe especificarse MQOO_OUTPUT.
6. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT.
7. Esta autorización es necesaria tanto para el objeto gestor de colas como para la cola concreta.
8. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT y MQOO_SET_IDENTITY_CONTEXT.
9. Esta comprobación se realiza para una cola local o modelo cuyo atributo de cola *Usage* sea MQUS_TRANSMISSION y se esté abriendo directamente para salida. Esto no es aplicable si se abre una cola remota (especificando los nombres del gestor de colas remoto y la cola remota, o especificando el nombre de una definición local de la cola remota).
10. También debe especificarse como mínimo una de las opciones MQOO_INQUIRE (para cualquier tipo de objeto) o (para colas) MQOO_BROWSE, MQOO_INPUT_*, MQOO_OUTPUT o MQOO_SET. La comprobación que se lleva a cabo es la misma que en las otras opciones especificadas, utilizando el identificador de usuario alternativo suministrado para la autorización sobre el objeto específico nombrado, y la autorización sobre la aplicación actual para la comprobación MQZAO_ALTERNATE_USER_IDENTIFIER.
11. Esta autorización permite especificar cualquier *AlternateUserId*.
12. También se realiza una comprobación MQZAO_OUTPUT si la cola no tiene un atributo de cola *Usage* de MQUS_TRANSMISSION.
13. La comprobación que se lleva a cabo es la misma que en las otras opciones especificadas, utilizando el identificador de usuario alternativo suministrado para la autorización sobre la cola nombrada, y la autorización sobre la aplicación actual para la comprobación MQZAO_ALTERNATE_USER_IDENTIFIER.
14. La comprobación solo se lleva a cabo si se cumplen las dos sentencias siguientes:
    - Se está cerrando y suprimiendo una cola dinámica permanente.
    - La cola no ha sido creado por la llamada a MQOPEN que ha devuelto el descriptor de contexto de objeto que se utiliza.

    De lo contrario, no hay comprobación.

Notas generales:

1. La autorización especial MQZAO_ALL_MQI incluye todas las autorizaciones siguientes que sean aplicables al tipo de objeto:
   - MQZAO_CONNECT
   - MQZAO_INQUIRE
   - MQZAO_SET
   - MQZAO_BROWSE
   - MQZAO_INPUT
   - MQZAO_OUTPUT
   - MQZAO_PASS_IDENTITY_CONTEXT
   - MQZAO_PASS_ALL_CONTEXT
   - MQZAO_SET_ IDENTITY_ CONTEXT
   - MQZAO_SET_ ALL_CONTEXT
   - MQZAO_ALTERNATE_USER_AUTHORITY
2. MQZAO_DELETE (vea la nota "14" en la página 169) y MQZAO_DISPLAY están clasificadas como autorizaciones de administración. Por lo tanto no están incluidas en MQZAO_ALL_MQI.
3. *No se comprueba* significa que no se lleva a cabo la comprobación.
4. *No es aplicable* significa que la comprobación de autorización no tiene sentido en esta operación. Por ejemplo, no se puede emitir una llamada MQPUT dirigida a un objeto proceso.

IBM i *Autorizaciones para mandatos MQSC en los PCF de escape en IBM i*

Estas autorizaciones permiten a un usuario emitir mandatos de administración como un mensaje PCF de escape. Estos métodos permiten a un programa enviar un mandato de administración como un mensaje a un gestor de colas, para que se ejecute en nombre de dicho usuario.

Esta sección resume las autorizaciones necesarias para cada mandato MQSC contenido en un PCF de escape.

No es aplicable significa que la comprobación de autorización no tiene sentido en esta operación.

El ID de usuario bajo el que se ejecuta el programa que envía el mandato también debe tener las autorizaciones siguientes:

- Autorización MQZAO_CONNECT para el gestor de colas
- Autorización DISPLAY sobre el gestor de colas para realizar mandatos PCF
- Autorización para emitir mandatos MQSC dentro del texto del mandato PCF de escape

ALTER *objeto*

Objeto	**Autorización necesaria**
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	MQZAO_CHANGE
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE

Objeto	Autorización necesaria
Servicio	MQZAO_CHANGE

CLEAR *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CLEAR
Tema	MQZAO_CLEAR
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	No aplicable
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

DEFINE *objeto* NOREPLACE (“1” en la página 174)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(“2” en la página 174)**
Tema	MQZAO_CREATE **(“2” en la página 174)**
Proceso	MQZAO_CREATE **(“2” en la página 174)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(“2” en la página 174)**
Información de autenticación	MQZAO_CREATE **(“2” en la página 174)**
Canal	MQZAO_CREATE **(“2” en la página 174)**
Canal de conexión de cliente	MQZAO_CREATE **(“2” en la página 174)**
Escucha	MQZAO_CREATE **(“2” en la página 174)**
Servicio	MQZAO_CREATE **(“2” en la página 174)**

DEFINE *objeto* REPLACE (“1” en la página 174, “3” en la página 174)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE

Objeto	Autorización necesaria
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE

DELETE *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DELETE
Tema	MQZAO_DELETE
Proceso	MQZAO_DELETE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_DELETE
Información de autenticación	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexión de cliente	MQZAO_DELETE
Escucha	MQZAO_DELETE
Servicio	MQZAO_DELETE

DISPLAY *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DISPLAY
Tema	MQZAO_DISPLAY
Proceso	MQZAO_DISPLAY
Gestor de colas	MQZAO_DISPLAY
Lista de nombres	MQZAO_DISPLAY
Información de autenticación	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexión de cliente	MQZAO_DISPLAY
Escucha	
Servicio	

PING CHANNEL

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable

Objeto	Autorización necesaria
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

RESET CHANNEL

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL_EXTENDED
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

RESOLVE CHANNEL

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL_EXTENDED
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

START *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable

Objeto	Autorización necesaria
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL

STOP *objeto*

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	MQZAO_CONTROL
Servicio	MQZAO_CONTROL

Nota:

1. Para los mandatos DEFINE, se necesita también la autorización MQZAO_DISPLAY sobre el objeto LIKE, si se ha especificado uno, o sobre el objeto SYSTEM.DEFAULT.xxx adecuado si se ha omitido LIKE.
2. La autorización MQZAO_CREATE no es específica de un objeto o tipo de objeto en particular. La autorización para crear se otorga sobre todos los objetos de un gestor de colas indicado, especificando el tipo de objeto QMGR en el mandato GRTMQMAUT.
3. Esta opción es aplicable si el objeto que va a sustituirse ya existe. Si no existe, la comprobación es como para DEFINE *objeto* NOREPLACE.

IBM i *Autorizaciones para mandatos PCF en IBM i*

Estas autorizaciones permiten a un usuario emitir mandatos de administración como mandatos PCF. Estos métodos permiten a un programa enviar un mandato de administración como un mensaje a un gestor de colas, para que se ejecute en nombre de dicho usuario.

Esta sección resume las autorizaciones necesarias para cada mandato PCF.

La indicación *No se comprueba* significa que no se lleva a cabo ninguna comprobación de autorización; *No aplicable* significa que la comprobación de autorización no es pertinente en esta operación.

El ID de usuario bajo el que se ejecuta el programa que envía el mandato también debe tener las autorizaciones siguientes:

- Autorización MQZAO_CONNECT para el gestor de colas
- Autorización DISPLAY sobre el gestor de colas para realizar mandatos PCF

La autorización especial MQZAO_ALL_ADMIN incluye las siguientes autorizaciones:

- MQZAO_CHANGE
- MQZAO_CLEAR
- MQZAO_DELETE
- MQZAO_DISPLAY
- MQZAO_CONTROL
- MQZAO_CONTROL_EXTENDED

MQZAO_CREATE no está incluida porque no es específica de un objeto o tipo de objeto en particular.

Change *objeto*

Objeto	**Autorización necesaria**
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	MQZAO_CHANGE
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE

Clear *objeto*

Objeto	**Autorización necesaria**
Cola	MQZAO_CLEAR
Tema	MQZAO_CLEAR
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	No aplicable
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Copy *objeto* (without replace) (“1” en la página 180)

Objeto	**Autorización necesaria**
Cola	MQZAO_CREATE **(“2” en la página 180)**
Tema	MQZAO_CREATE **(“2” en la página 180)**

Objeto	Autorización necesaria
Proceso	MQZAO_CREATE **(“2” en la página 180)**
Gestor de colas	No aplicable
NamelistMQZAO_CREATE	MQZAO_CREATE **(“2” en la página 180)**
Información de autenticación	MQZAO_CREATE **(“2” en la página 180)**
Canal	MQZAO_CREATE **(“2” en la página 180)**
Canal de conexión de cliente	MQZAO_CREATE **(“2” en la página 180)**
Escucha	MQZAO_CREATE **(“2” en la página 180)**
Servicio	MQZAO_CREATE **(“2” en la página 180)**

Copy *objeto* (with replace) (“1” en la página 180, “4” en la página 180)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE

Create *objeto* (without replace) (“3” en la página 180)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(“2” en la página 180)**
Tema	MQZAO_CREATE **(“2” en la página 180)**
Proceso	MQZAO_CREATE **(“2” en la página 180)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(“2” en la página 180)**
Información de autenticación	MQZAO_CREATE **(“2” en la página 180)**
Canal	MQZAO_CREATE **(“2” en la página 180)**
Canal de conexión de cliente	MQZAO_CREATE **(“2” en la página 180)**
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE

Create *objeto* (with replace) (“3” en la página 180, “4” en la página 180)

Objeto	Autorización necesaria
Cola	MQZAO_CHANGE
Tema	MQZAO_CHANGE
Proceso	MQZAO_CHANGE
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CHANGE
Información de autenticación	MQZAO_CHANGE
Canal	MQZAO_CHANGE
Canal de conexión de cliente	MQZAO_CHANGE
Escucha	MQZAO_CHANGE
Servicio	MQZAO_CHANGE

Delete *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DELETE
Tema	MQZAO_DELETE
Proceso	MQZAO_DELETE
Gestor de colas	MQZAO_DELETE
Lista de nombres	MQZAO_DELETE
Información de autenticación	MQZAO_DELETE
Canal	MQZAO_DELETE
Canal de conexión de cliente	MQZAO_DELETE
Escucha	MQZAO_DELETE
Servicio	MQZAO_DELETE

Inquire *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_DISPLAY
Tema	MQZAO_DISPLAY
Proceso	MQZAO_DISPLAY
Gestor de colas	MQZAO_DISPLAY
Lista de nombres	MQZAO_DISPLAY
Información de autenticación	MQZAO_DISPLAY
Canal	MQZAO_DISPLAY
Canal de conexión de cliente	MQZAO_DISPLAY
Escucha	MQZAO_DISPLAY

Objeto	Autorización necesaria
Servicio	MQZAO_DISPLAY

Inquire *objeto* names

Objeto	Autorización necesaria
Cola	No se comprueba
Tema	No se comprueba
Proceso	No se comprueba
Gestor de colas	No se comprueba
Lista de nombres	No se comprueba
Información de autenticación	No se comprueba
Canal	No se comprueba
Canal de conexión de cliente	No se comprueba
Escucha	No se comprueba
Servicio	No se comprueba

Sondear canal

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Restablecer canal

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL_EXTENDED

Objeto	Autorización necesaria
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Restablecer estadísticas de la cola

Objeto	Autorización necesaria
Cola	MQZAO_DISPLAY y MQZAO_CHANGE
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	No aplicable
Canal de conexión de cliente	No aplicable
Escucha	
Servicio	

Resolver canal

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL_EXTENDED
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Iniciar canal

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable

Objeto	Autorización necesaria
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Detener canal

Objeto	Autorización necesaria
Cola	No aplicable
Tema	No aplicable
Proceso	No aplicable
Gestor de colas	No aplicable
Lista de nombres	No aplicable
Información de autenticación	No aplicable
Canal	MQZAO_CONTROL
Canal de conexión de cliente	No aplicable
Escucha	No aplicable
Servicio	No aplicable

Nota:

1. En los mandatos Copy, también es necesaria la autorización MQZAO_DISPLAY para el objeto de origen.
2. La autorización MQZAO_CREATE no es específica de un objeto o tipo de objeto en particular. La autorización para crear se otorga sobre todos los objetos de un gestor de colas indicado, especificando el tipo de objeto QMGR en el mandato GRTMQMAUT.
3. Para los mandatos Create, también se necesita la autorización MQZAO_DISPLAY para el SYSTEM.DEFAULT.* .
4. Esta opción es aplicable si el objeto que va a sustituirse ya existe. Si no existe, la comprobación es como para un mandato Copy o Create sin sustitución.

IBM i Perfiles OAM genéricos en IBM i

Los perfiles genéricos del gestor de autorizaciones sobre objetos (OAM) le permiten establecer de una sola vez la autorización que un usuario tiene sobre muchos objetos, en lugar de tener que emitir mandatos **GRTMQMAUT** distintos para cada objeto individual en el momento de su creación. La utilización de perfiles genéricos en el mandato **GRTMQMAUT** permite establecer una autorización genérica para todos los objetos que se creen en el futuro que se ajusten a dicho perfil.

El resto de esta sección describe con más detalle el uso de los perfiles genéricos:

- “Utilización de caracteres comodín” en la página 180
- “Prioridades de perfiles” en la página 181

Utilización de caracteres comodín

Lo que hace que un perfil sea genérico es el uso de caracteres especiales (caracteres comodín) en el nombre del perfil. Por ejemplo, el carácter comodín de signo de interrogación (?) coincide con cualquier

carácter individual de un nombre. Por tanto, si se especifica ABC.?EF, la autorización que se otorga a dicho perfil se aplica a todos los objetos creados con los nombres ABC.DEF, ABC.CEF, ABC.BEF, etcétera.

Los caracteres comodín disponibles son:

?

Utilice el signo de interrogación (?) en lugar de cualquier otro carácter. Por ejemplo, AB.?D se aplicaría a los objetos AB.CD, AB.EDy AB.FD.

*****

Utilice el asterisco (*) como:

- Un *calificador* de un nombre de perfil para que coincida con cualquier calificador de un nombre de objeto. Un calificador es la parte de un nombre de objeto delimitada por un punto. Por ejemplo, en ABC.DEF.GHI, los calificadores son ABC, DEF y GHI.

  Por ejemplo, ABC.*.JKL se aplicaría a los objetos ABC.DEF.JKLy ABC.GHI.JKL. Tenga en cuenta que **no** se aplicará ABC.JKL. Cuando se utiliza el carácter * en este contexto siempre indica un calificador.
- Un carácter contenido en un calificador de un nombre de perfil para que coincida con cero o más caracteres incluidos en el calificador de un nombre de objeto.

  Por ejemplo, ABC.DE*.JKL se aplicaría a los objetos ABC.DE.JKL, ABC.DEF.JKLy ABC.DEGH.JKL.

******

Utilice el asterisco doble (**) ***una vez*** en el nombre de un perfil como:

- El nombre de perfil completo para que coincida con todos los nombres de objetos. Por ejemplo, si utiliza la palabra clave OBJTYPE (*PRC) para identificar procesos y luego utiliza ** como el nombre del perfil, se cambian las autorizaciones de todos los procesos.
- Como cualquier calificador del principio, mitad o final de un nombre de perfil para que coincida con cero o más calificadores contenidos en un nombre de objeto. Por ejemplo, **.ABC identifica todos los objetos con el calificador final ABC.

Prioridades de perfiles

Una cuestión importante que debe comprender cuando utilice perfiles genéricos es la prioridad que se otorga a los perfiles a la hora de decidir qué autorizaciones se han de aplicar a un objeto que se está creando. Por ejemplo, suponga que ha emitido los mandatos:

```
GRTMQMAUT OBJ(AB.*) OBJTYPE(*Q) USER(FRED) AUT(*PUT) MQMNAME(MYQMGR)
GRTMQMAUT OBJ(AB.C*) OBJTYPE(*Q) USER(FRED) AUT(*GET) MQMNAME(MYQMGR)
```

La primera otorga autorización de colocación a todas las colas para el principal FRED con nombres que coinciden con el perfil AB. *; el segundo otorga autorización de obtención sobre los mismos tipos de cola que coinciden con el perfil AB.C*.

Suponga que ahora crea una cola con el nombre AB.CD. De acuerdo con las reglas para las comparaciones con comodines, cualquiera de los dos mandatos GRTMQMAUT podría aplicarse a dicha cola. Por lo tanto, la cuestión es ¿tiene autorización para transferir o para obtener?

Para encontrar la respuesta, aplique la regla según la cual cuando varios perfiles pueden aplicarse a un objeto, **sólo se aplica el más específico**. El modo en que se aplica esta regla es comparando los nombres de perfil de izquierda a derecha. Siempre que difieren, un carácter no genérico es más especifico que un carácter genérico. De este modo, en el ejemplo anterior, la cola AB.CD tiene autorización para **obtener** (AB.C* es más específico que AB.*).

Cuando se comparan caracteres genéricos, el orden de *especificidad* es el siguiente:

1. ?
2. *

3. **

IBM i Especificación del servicio de autorización instalado en IBM i

Puede especificar el componente de servicio de autorización que se ha de utilizar.

El parámetro **Service Component name** en **GRTMQMAUT** y **RVKMQMAUT** le permite especificar el nombre del componente de servicio de autorización instalado.

Seleccionar **F24** en el panel inicial, seguido de **F9=Todos los parámetros** en el siguiente panel de cualquiera de los dos mandatos, permite especificar el componente de autorización instalado (*DFT) o bien el nombre del componente de servicio de autorización necesario especificado en la sección Service del archivo qm.ini del gestor de colas.

DSPMQMAUT también tiene este parámetro adicional. Este parámetro permite buscar el nombre de objeto, el tipo de objeto y el usuario especificados en todos los componentes de autorización instalados (*DFT), o bien en el nombre del componente de servicio de autorización especificado.

IBM i Trabajar con y sin perfiles de autorización en IBM i

Utilice esta información para aprender a trabajar con perfiles de autorización y a trabajar sin perfiles de autorización.

Puede trabajar con perfiles de autorización, como se explica en "Trabajar con perfiles de autorización" en la página 182, o sin ellos, como se explica a continuación:

Para trabajar sin perfiles de autorización, utilice *NONE como parámetro Authority en **GRTMQMAUT** para crear perfiles sin autorización. Los perfiles existentes se quedan igual.

En **RVKMQMAUT**, utilice *REMOVE como parámetro Authority para eliminar un perfil de autorización existente.

Trabajar con perfiles de autorización

Hay dos mandatos asociados con la creación de perfiles de autorización:

- **WRKMQMAUT**
- **WRKMQMAUTD**

Puede acceder a estos mandatos directamente desde la línea de mandatos o bien desde el panel WRKMQM realizando lo siguiente:

1. Escriba el nombre del gestor de colas y pulse la tecla Enter para acceder al panel de resultados de **WRKMQM** .
2. Seleccionando F23=More options en este panel.

La opción 24 selecciona el panel de resultados para el mandato **WRKMQMAUT** y la opción 25 selecciona el mandato **WRKMQMAUTI** , que se utiliza con la capa de enlaces SSL.

WRKMQMAUT

Este mandato permite trabajar con los datos de autorizaciones que se guardan en la cola de autorizaciones.

Nota: Para ejecutar este mandato, debe tener autorización *connect y *admdsp sobre el gestor de colas. Sin embargo, para crear o suprimir un perfil, necesita la autorización QMQMADM.

Si envía la información a la pantalla, se visualiza una lista de nombres de perfiles de autorización junto con sus tipos. Si imprime la salida, recibirá una lista detallada de todos los datos de autorizaciones, los usuarios registrados y sus autorizaciones.

Al especificar un nombre de objeto o perfil en este panel, y al pulsar INTRO se le lleva al panel de resultados de **WRKMQMAUT** .

Si selecciona 4=Delete, vaya a un nuevo panel desde el que puede confirmar que desea suprimir todos los nombres de usuario registrados en el nombre de perfil de autorización genérico que especifique. Esta opción ejecuta el mandato **RVKMQMAUT** con la opción *REMOVE para todos los usuarios y **sólo** se aplica a los nombres de perfiles genéricos.

Si selecciona 12=Work with profile , vaya al panel de resultados del mandato **WRKMQMAUTD** , tal como se explica en "WRKMQMAUTD" en la página 183.

WRKMQMAUTD

Este mandato permite visualizar todos los usuarios registrados con un nombre de perfil de autorización y un tipo de objeto determinados. Para ejecutar este mandato, debe tener autorización *connect y *admdsp sobre el gestor de colas. Sin embargo, para otorgar, ejecutar, crear o suprimir un perfil, necesita la autorización QMQMADM.

Al seleccionar F24=More keys en el panel de entrada inicial, seguido de la opción F9=All Parameters se muestra el nombre de componente de servicio como para **GRTMQMAUT** y **RVKMQMAUT**.

Nota: La clave F11=Display Object Authorizations conmuta entre los siguientes tipos de autorizaciones:

- Autorizaciones de objetos
- Autorizaciones de contexto
- Autorizaciones de MQI

Las opciones que aparecen en la pantalla son:

2=Grant
Le lleva al panel del mandato **GRTMQMAUT** para añadirlo a las autorizaciones actuales.

3=Revoke
Le lleva al panel **RVKMQMAUT** para eliminar algunas de las definiciones actuales.

4=Delete
Le lleva a un panel que le permite suprimir los datos de autorizaciones de usuarios especificados. Ejecuta el mandato **RVKMQMAUT** con la opción *REMOVE.

5=Display
Le lleva al mandato **DSPMQMAUT** existente.

F6=Create
Le lleva al panel de **GRTMQMAUT** que le permite crear un registro de autorización de perfil.

IBM i Directrices para el Gestor de autorizaciones sobre objetos (OAM) en IBM i

Consejos y sugerencias adicionales para utilizar el gestor de autorizaciones sobre objetos (OAM)

Limitar el acceso a operaciones confidenciales

Algunas operaciones son confidenciales; limítelas a los usuarios con privilegios. Por ejemplo,

- Acceder a algunas colas especiales, tales como colas de transmisión o la cola de mandatos SYSTEM.ADMIN.COMMAND.QUEUE
- La ejecución de programas que utilicen todas las opciones de contexto de la MQI
- Crear y copiar colas de aplicación

Directorios del gestor de colas

Los directorios y las bibliotecas que contiene las colas y otros datos de los gestores de colas son privados del producto. No utilice mandatos del sistema operativo estándar para otorgar o revocar autorizaciones sobre recursos de la MQI.

Colas

La autorización sobre una cola dinámica se basa en la cola modelo de la que se deriva, aunque no tiene por qué ser igual.

En las colas alias y las colas remotas, la autorización es la del objeto propiamente dicho, no la de la cola en la que se resuelve la cola alias o remota. Es posible otorgar a un perfil de usuario autorización para que acceda a una cola alias que se resuelve en una cola local sobre la que el perfil de usuario no tenga permiso de acceso.

Limite la autorización para crear colas a los usuarios con privilegios. Si no lo hace así, los usuarios pueden eludir el control de acceso normal creando un alias.

Autorización de usuario alternativo

La autorización de usuario alternativo controla si un perfil de usuario puede utilizar la autorización de otro perfil de usuario al acceder a un objeto IBM MQ. Esta técnica es esencial cuando un servidor recibe solicitudes de un programa y el servidor desea asegurarse que el programa tiene la autorización necesaria para la solicitud. El servidor puede tener la autorización necesaria, pero necesita saber si el programa tiene autorización para las acciones que ha solicitado.

Por ejemplo:

- Un programa servidor que se está ejecutando bajo el perfil PAYSERV recupera de una cola un mensaje de solicitud que el perfil de usuario USER1 puso en la cola.
- Cuando el programa servidor obtiene el mensaje de solicitud, procesa la solicitud y vuelve a transferir la respuesta a la cola de respuestas especificada con el mensaje de solicitud.
- En lugar de utilizar su propio perfil de usuario (PAYSERV) para autorizar la apertura de una cola de respuestas, el servidor puede especificar algún otro perfil de usuario, en este caso, USER1. En este ejemplo, se puede emplear la autorización de usuario alternativo para controlar si PAYSERV tiene autorización para especificar USER1 como perfil de usuario alternativo al abrir la cola de respuestas.

El perfil de usuario alternativo se especifica en el campo *AlternateUserId* del descriptor de objeto.

Nota: Puede utilizar perfiles de usuario alternativo en cualquier objeto IBM MQ. El uso de un perfil de usuario alternativo no afecta al perfil de usuario utilizado por cualquier otro gestor de recursos.

Autorización de contexto

El contexto es la información que se aplica a un mensaje determinado y está contenida en el descriptor de mensaje, MQMD, que forma parte del mensaje.

Para obtener descripciones de los campos del descriptor de mensaje relacionados con el contexto, consulte Visión general de MQMD.

Para obtener información sobre las opciones de contexto, consulte Contexto de mensaje.

Consideraciones sobre la seguridad remota

Para la seguridad remota, tenga en cuenta lo siguiente:

Autoridad de transferencia
Por razones de seguridad entre los gestores de colas, puede especificar la autorización para transferir que se utiliza cuando un canal recibe un mensaje enviado desde otro gestor de colas.

Este parámetro sólo es válido para tipos de canal RCVR, RQSTR o CLUSRCVR. Especifique el atributo de canal PUTAUT como se indica a continuación:

DEF
Perfil de usuario predeterminado. Es el perfil de usuario QMQM bajo el que se está ejecutando el agente de canal de mensajes.

CTX
El perfil de usuario del contexto de mensaje.

Colas de transmisión
Los gestores de colas transfieren automáticamente los mensajes remotos a una cola de transmisión; no se requiere ninguna autorización especial. Sin embargo, se necesita una autorización especial para transferir un mensaje directamente a una cola de transmisión.

Salidas de canal
Las salidas de canal se pueden utilizar para implementar medidas de seguridad adicionales.

Registros de autenticación de canal
Se utiliza para ejercer un control más preciso sobre el acceso otorgado a la conexión de sistemas a nivel de canal.

Si desea más información sobre la seguridad remota, consulte "Autorización de canal" en la página 110.

Protección de canales con SSL/TLS

El protocolo TLS (seguridad de la capa de transporte) proporciona seguridad de canal, con protección contra escuchas y manipulaciones no autorizadas y contra falsas identidades. El soporte de IBM MQ para TLS le permite especificar, en la definición de canal, que un canal determinado utilice seguridad TLS. También puede especificar detalles de la seguridad que desea, como por ejemplo el algoritmo de cifrado que desea utilizar.

El soporte de TLS en IBM MQ utiliza el *objeto de información de autenticación* del gestor de colas y diversos mandatos CL y MQSC, así como parámetros de gestor de colas y canal que definen detalladamente el soporte de TLS necesario.

Los siguientes mandatos CL tienen soporte para TLS:

WRKMQMAUTI
Trabajar con los atributos de un objeto de información de autenticación.

CHGMQMAUTI
Modificar los atributos de un objeto de información de autenticación.

CRTMQMAUTI
Crear un objeto de información de autenticación.

CPYMQMAUTI
Crear un objeto de información de autenticación copiando uno existente.

DLTMQMAUTI
Suprimir un objeto de información de autenticación.

DSPMQMAUTI
Visualiza los atributos de un objeto de información de autenticación específico.

Para obtener una visión general de la seguridad de canal utilizando TLS, consulte

- Protección de los canales con TLS

Para conocer detalles de los mandatos PCF asociados a TLS, consulte

- Cambiar, copiar y crear un objeto de información de autenticación
- Suprimir objeto de información de autenticación
- Consultar objeto de información de autenticación

z/OS Configuración de la seguridad en z/OS

Consideraciones sobre seguridad específicas de z/OS.

Seguridad en IBM MQ for z/OS se controla mediante RACF o un gestor de seguridad externa (ESM) equivalente.

En las instrucciones siguientes se presupone que utiliza RACF.

Referencia relacionada
Escenario de seguridad: dos gestores de colas en z/OS

z/OS Clases de seguridad de RACF

Las clases RACF se utilizan para contener los perfiles necesarios para la comprobación de seguridad de IBM MQ. Muchas de las clases de miembro tienen clases de grupo equivalentes. Debe activar las clases y habilitarlas para que acepten perfiles genéricos.

Cada clase RACF contiene uno o más perfiles que se utilizan en algún momento en la secuencia de comprobación, tal como se muestra en la Tabla 23 en la página 186.

Tabla 23. Clases RACF utilizadas por IBM MQ

Clase de miembro	Clase de grupo	Contenido
MQADMIN	GMQADMIN	Perfiles: Se utilizan principalmente para contener perfiles para funciones de tipo administrativo. Por ejemplo: • Perfiles para conmutadores de seguridad de IBM MQ • El perfil de seguridad RESLEVEL • Perfiles para la seguridad de usuario alternativo • El perfil de seguridad de contexto • Perfiles para la seguridad de recursos de mandatos
MXADMIN	GMXADMIN	Perfiles: Se utilizan principalmente para contener perfiles para funciones de tipo administrativo. Por ejemplo: • Perfiles para conmutadores de seguridad de IBM MQ • El perfil de seguridad RESLEVEL • Perfiles para la seguridad de usuario alternativo • El perfil de seguridad de contexto • Perfiles para la seguridad de recursos de mandatos Esta clase puede contener perfiles RACF en mayúsculas y en una combinación de mayúsculas y minúsculas.
MQCONN		Perfiles utilizados para la seguridad de conexión
MQCMDS		Perfiles utilizados para la seguridad de mandatos
MQQUEUE	GMQQUEUE	Perfiles utilizados en la seguridad de recursos de colas
MXQUEUE	GMXQUEUE	Perfiles en mayúsculas y en una combinación de mayúsculas y minúsculas utilizados en la seguridad de recursos de colas
MQPROC	GMQPROC	Perfiles utilizados en la seguridad de recursos de procesos
MXPROC	GMXPROC	Perfiles en mayúsculas y en una combinación de mayúsculas y minúsculas utilizados en la seguridad de recursos de procesos
MQNLIST	GMQNLIST	Perfiles utilizados en la seguridad de recursos de listas de nombres
MXNLIST	GMXNLIST	Perfiles en mayúsculas y en una combinación de mayúsculas y minúsculas utilizados en la seguridad de recursos de listas de nombres

Tabla 23. Clases RACF utilizadas por IBM MQ (continuación)		
Clase de miembro	**Clase de grupo**	**Contenido**
MXTOPIC	GMXTOPIC	Perfiles en mayúsculas y en una combinación de mayúsculas y minúsculas utilizados en la seguridad de temas

Algunas clases tienen una *clase de grupo* relacionada que le permite agrupar grupos de recursos que tienen requisitos de acceso similares. Para obtener detalles sobre la diferencia entre las clases de miembro y grupo y cuándo utilizar un miembro o clase de grupo, consulte la publicación z/OS Security Server RACF Security Administrator's Guide.

Las clases deben activarse antes de poder realizar las comprobaciones de seguridad. Para activar todas las clases de IBM MQ, puede utilizar este mandato RACF:

```
SETROPTS CLASSACT(MQADMIN,MXADMIN,MQQUEUE,MXQUEUE,MQPROC,MXPROC,
                  MQNLIST,MXNLIST,MXTOPIC,MQCONN,MQCMDS)
```

Asimismo, debería asegurarse de configurar las clases para que puedan aceptar perfiles genéricos. También puede hacerlo con el mandato SETROPTS de RACF, por ejemplo:

```
SETROPTS GENERIC(MQADMIN,MXADMIN,MQQUEUE,MXQUEUE,MQPROC,MXPROC,
                 MQNLIST,MXNLIST,MXTOPIC,MQCONN,MQCMDS)
```

z/OS Perfiles RACF

Todos los perfiles RACF que utiliza IBM MQ contienen un prefijo, que es el nombre del gestor de colas o el nombre del grupo de compartición de colas. Tenga cuidado cuando utilice el signo de tanto por ciento como carácter comodín.

Todos los perfiles RACF utilizados por IBM MQ contienen un perfil. Para la seguridad a nivel de grupo de compartición de colas, el prefijo es el nombre del grupo de compartición de colas. Para la seguridad a nivel de gestor de colas, es el nombre del gestor de colas. Si está utilizando una combinación de seguridad a nivel de gestor de colas y de grupo de compartición de colas, utilizará perfiles con los dos tipos de prefijo. (El grupo de compartición de colas y la seguridad a nivel de gestor de colas se describen en IBM MQ for z/OSConceptos: seguridad.)

Por ejemplo, si desea proteger una cola llamada QUEUE_FOR_SUBSCRIBER_LIST en el grupo de compartición de colas QSG1 a nivel de grupo de compartición de colas, el perfil adecuado se definiría en RACF como:

```
RDEFINE MQQUEUE QSG1.QUEUE_FOR_SUBSCRIBER_LIST
```

Si desea proteger una cola llamada QUEUE_FOR_LOST_CARD_LIST, que pertenece al gestor de colas STCD, a nivel de gestor de colas, el perfil adecuado se definiría en RACF como:

```
RDEFINE MQQUEUE STCD.QUEUE_FOR_LOST_CARD_LIST
```

Esto significa que diferentes gestores de colas y grupos de compartición de colas pueden compartir la misma base de datos RACF y, sin embargo, tener opciones de seguridad diferentes.

No utilice nombres de gestor de colas genéricos en los perfiles para evitar un acceso de usuarios imprevisto.

IBM MQ permite el uso del signo de tanto por ciento (%) en nombres de objeto. Sin embargo, RACF utiliza el carácter % como comodín de único carácter. Esto significa que cuando defina un nombre de objeto que contenga un carácter % en el nombre, debe tener esto en cuenta cuando defina el perfil correspondiente.

Por ejemplo, para la cola CREDIT_CARD_%_RATE_INQUIRY, en el gestor de colas CRDP, el perfil se definiría en RACF de la siguiente manera:

```
RDEFINE MQQUEUE CRDP.CREDIT_CARD_%_RATE_INQUIRY
```

Esta cola no se puede proteger con un perfil genérico, como por ejemplo CRDP.**.

IBM MQ permite el uso de caracteres en mayúsculas y minúsculas en nombres de objeto. Puede proteger estos objetos definiendo:

1. Perfiles que combinan mayúsculas y minúsculas en las clases RACF que combinan mayúsculas y minúsculas adecuadas o
2. Perfiles genéricos en las clases RACF en mayúsculas adecuadas.

Para utilizar perfiles de mayúsculas y minúsculas y clases RACF de mayúsculas y minúsculas, debe seguir los pasos descritos en "z/OS Migración de un gestor de colas a la seguridad de mayúsculas y minúsculas" en la página 272.

Hay algunos perfiles, o partes de perfiles, que permanecen en mayúsculas solamente ya que los valores los proporciona IBM MQ. Son las siguientes:

- Perfiles de conmutador.
- Todos los calificadores de alto nivel (HLQ) incluyendo los identificadores de subsistema y de grupo de compartición de colas.
- Perfiles para objetos SYSTEM.
- Perfiles para objetos predeterminados.
- La clase **MQCMDS**, por lo que todos los perfiles de mandato son en mayúsculas solamente.
- La clase **MQCONN**, por lo que todos los perfiles de conexión son en mayúsculas solamente.
- Perfiles **RESLEVEL**.
- La calificación `'object'` en perfiles de recurso de mandato; por ejemplo, `hlq.QUEUE.queuename`. Sólo el nombre de recurso está en una combinación de mayúsculas y minúsculas.
- Perfiles de cola dinámica `hlq.CSQOREXX.*`, `hlq.CSQUTIL.*` y `CSQXCMD.*`.
- La parte `'CONTEXT'` de `hlq.CONTEXT.resourcename`.
- La parte `'ALTERNATE.USER'` de `hlq.ALTERNATE.USER.userid`.

Por ejemplo, si tiene una cola llamada `PAYROLL.Dept1` en el gestor de colas `QM01` y está utilizando:

- Perfiles que combinan mayúsculas y minúsculas; puede definir un perfil en la clase IBM MQ RACF MXQUEUE

  ```
  RDEFINE MXQUEUE MQ01.PAYROLL.Dept1
  ```

- Perfiles en mayúsculas; puede definir un perfil en la clase IBM MQ RACF MQQUEUE

  ```
  RDEFINE MQQUEUE MQ01.PAYROLL.*
  ```

El primer ejemplo, en el que se utilizan perfiles que combinan mayúsculas y minúsculas, le ofrece un control más detallado sobre la concesión de autorización para acceder al recurso.

z/OS Perfiles de conmutador

Para controlar la comprobación de seguridad realizada por IBM MQ, se utilizan *perfiles de conmutador*. Un perfil de conmutador es un perfil RACF normal que tiene un significado especial para IBM MQ. IBM MQ no utiliza la lista de acceso de los perfiles de conmutador.

IBM MQ mantiene un conmutador interno para cada tipo de conmutador mostrado en las tablas Perfiles de conmutador para seguridad a nivel de subsistema, Perfiles de conmutador para seguridad a nivel de grupo de compartición de colas o de gestor de colas y Perfiles de conmutador para comprobación de recursos. Los perfiles de conmutador se pueden mantener a nivel de grupo de compartición de colas, a nivel de gestor de colas o en una combinación de ambos. Mediante la utilización de un único conjunto de perfiles de conmutador de seguridad de grupo de compartición de colas, puede controlar la seguridad en todos los gestores de colas de un grupo de compartición de colas.

Cuando un conmutador de seguridad está activado, se realizan las comprobaciones de seguridad asociadas al conmutador. Cuando un conmutador de seguridad está desactivado, las comprobaciones asociadas al conmutador se pasan por alto. El valor predeterminado es que todos los conmutadores de seguridad estén activados.

z/OS *Conmutadores y clases*

Cuando se inicia un gestor de colas o se renueva la seguridad, IBM MQ establece conmutadores en función del estado de varias clases RACF.

Cuando se inicia un gestor de colas (o cuando se renueva la clase MQADMIN o MXADMIN mediante el mandato IBM MQ REFRESH SECURITY), IBM MQ primero comprueba el estado de RACF y la clase adecuada:

- La clase MQADMIN, si está utilizando perfiles en mayúsculas
- La clase MXADMIN, si está utilizando perfiles que combinan mayúsculas y minúsculas.

Desactiva el conmutador de seguridad de subsistema si se cumple cualquiera de estas condiciones:

- RACF está inactivo o no está instalado.
- La clase MQADMIN o MXADMIN no está definida (estas clases siempre se definen para RACF porque se incluyen en la tabla de descriptores de clase (CDT)).
- La clase MQADMIN o MXADMIN no se ha activado.

Si tanto RACF como la clase MQADMIN o MXADMIN están activos, IBM MQ comprueba la clase MQADMIN o MXADMIN para ver si se ha definido alguno de los perfiles de conmutador. Primero comprueba los perfiles descritos en “Perfiles para controlar la seguridad de subsistema” en la página 190. Si la seguridad de subsistema no es necesaria, IBM MQ desactiva el conmutador interno de seguridad de subsistema y no realiza ninguna otra comprobación.

Los perfiles determinan si el conmutador de IBM MQ correspondiente está activado o desactivado.

- Si el conmutador está desactivado, dicho tipo de seguridad está desactivado.
- Si algún conmutador de IBM MQ está activado, IBM MQ comprueba el estado de la clase RACF asociada al tipo de seguridad correspondiente al conmutador de IBM MQ. Si la clase no está instalada o no está activa, el conmutador IBM MQ se desactiva. Por ejemplo, las comprobaciones de seguridad de procesos no se llevan a cabo si la clase MQPROC o MXPROC no se ha activado. Que la clase no esté activa equivale a definir un perfil NO.PROCESS.CHECKS para cada gestor de colas y grupo de compartición de colas que utiliza esta base RACF.

z/OS *Cómo funcionan los conmutadores*

Para desactivar un conmutador de seguridad, defina un NO.* perfil de conmutador para el mismo. Puede alterar temporalmente un NO.* perfil establecido a nivel de grupo de compartición de colas definiendo un archivo YES.* perfil para un gestor de colas.

Para desactivar un conmutador de seguridad, debe definir un NO.* perfil de conmutador para el mismo. La existencia de un NO.* El perfil significa que las comprobaciones de seguridad **no** se realizan para ese tipo

de recurso, a menos que elija alterar temporalmente un valor de nivel de grupo de compartición de colas en un gestor de colas determinado. Este tema se describe en el apartado “Alteración temporal de valores a nivel de grupo de compartición de colas” en la página 190.

Si su gestor de colas no es miembro de un grupo de compartición de colas, no necesita definir ningún perfil a nivel de grupo de compartición de colas ni ningún perfil de alteración temporal. Sin embargo, recuerde que debe definir estos perfiles si el gestor de colas se une a un grupo de compartición de colas posteriormente.

Cada NO.* perfil de conmutador que IBM MQ detecta desactiva la comprobación de ese tipo de recurso. Los perfiles de conmutador se activan durante el inicio del gestor de colas. Si cambia los perfiles de conmutador mientras alguno de los gestores de colas afectados está en ejecución, puede hacer que IBM MQ reconozca los cambios emitiendo el mandato IBM MQ REFRESH SECURITY.

Los perfiles de conmutador siempre deben definirse en la clase MQADMIN o MXADMIN. No los defina en la clase GMQADMIN o GMXADMIN. Las tablas Perfiles de conmutador para seguridad a nivel de subsistema y Perfiles de conmutador para comprobación de recursos muestran los perfiles de conmutador válidos y el tipo de seguridad que controlan.

Alteración temporal de valores a nivel de grupo de compartición de colas

Puede alterar temporalmente valores de seguridad a nivel de grupo de compartición de colas para un gestor de colas específico que sea miembro de dicho grupo. Si desea realizar comprobaciones de gestor de colas en un gestor de colas individual que no se realiza en otros gestores de colas del grupo, utilice (qmgr-name.YES. *) Perfiles de conmutador.

Por el contrario, si no desea realizar una determinada comprobación en un gestor de colas determinado dentro de un grupo de compartición de colas, defina un (qmgr-name.NO. *) perfil para ese tipo de recurso en particular en el gestor de colas y no definen un perfil para el grupo de compartición de colas. (IBM MQ sólo comprueba un perfil de nivel de grupo de compartición de colas si no encuentra un perfil de nivel de gestor de colas.)

z/OS *Perfiles para controlar la seguridad de subsistema*

IBM MQ comprueba si se requieren comprobaciones de seguridad de subsistema para el subsistema, para el gestor de colas y para el grupo de compartición de colas.

La primera comprobación de seguridad realizada por IBM MQ se utiliza para determinar si se requieren comprobaciones de seguridad para todo el subsistema IBM MQ. Si especifica que no desea seguridad de subsistema, no se realizan más comprobaciones.

Se comprueban los siguientes perfiles de conmutador para determinar si se requiere la seguridad de subsistema. La Figura 14 en la página 191 muestra el orden de comprobación.

Tabla 24. Perfiles de conmutador para seguridad a nivel de subsistema

Nombre del perfil de conmutador	**Tipo de recurso o comprobación que se controla**
qmgr-name.NO.SUBSYS.SECURITY	Seguridad de subsistema para este gestor de colas
qsg-name.NO.SUBSYS.SECURITY	Seguridad de subsistema para este grupo de compartición de colas
qmgr-name.YES.SUBSYS.SECURITY	Alteración temporal de la seguridad de subsistema para este gestor de colas

Si el gestor de colas no es miembro de un grupo de compartición de colas, IBM MQ solo comprueba la presencia del perfil de conmutador qmgr-name.NO.SUBSYS.SECURITY.

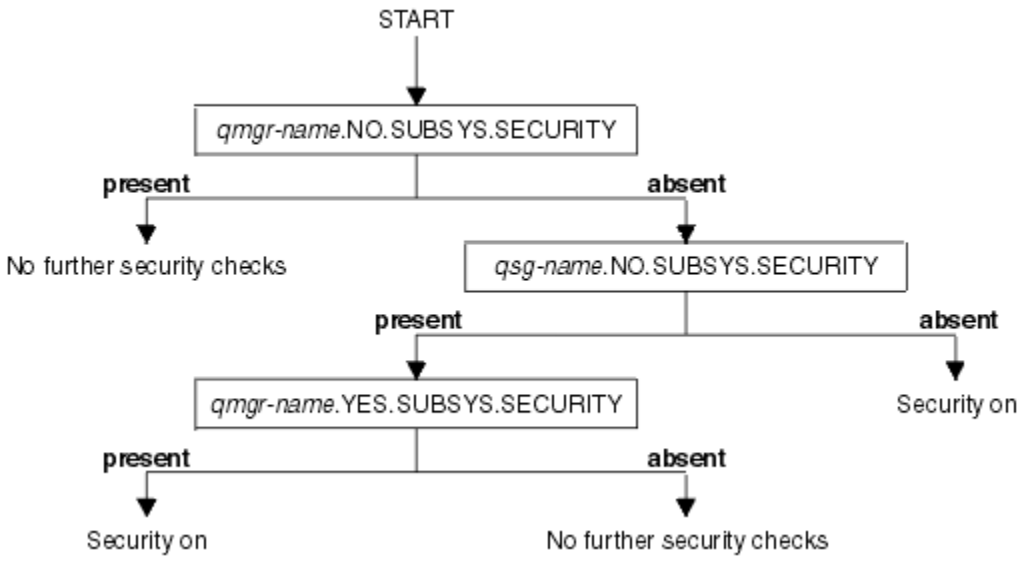


Figura 14. Comprobación de la seguridad de subsistema

z/OS *Perfiles para controlar la seguridad a nivel de grupo de compartición de colas o de gestor de colas*

Si es necesaria la comprobación de seguridad de subsistema, IBM MQ comprueba si se requiere comprobación de seguridad a nivel de grupo de compartición de colas o a nivel de gestor de colas.

Cuando IBM MQ ha determinado que es necesaria la comprobación de seguridad, determinará si la comprobación se requiere a nivel de grupo de compartición de colas o de gestor de colas, o en ambos. Estas comprobaciones no se realizan si el gestor de colas no es miembro de un grupo de compartición de colas.

Se comprueban los siguientes perfiles de conmutador para determinar el nivel necesario. La Figura 15 en la página 192 y la Figura 16 en la página 192 muestran el orden de comprobación.

Tabla 25. Perfiles de conmutador de seguridad a nivel de grupo de compartición de colas o de gestor de colas

Nombre del perfil de conmutador	Tipo de recurso o comprobación que se controla
qmgr-name.NO.QMGR.CHECKS	No se realizan comprobaciones a nivel de gestor de colas para este gestor de colas
qsg-name.NO.QMGR.CHECKS	No se realizan comprobaciones a nivel de gestor de colas para este grupo de compartición de colas
qmgr-name.YES.QMGR.CHECKS	Alteración temporal de las comprobaciones a nivel de gestor de colas para este gestor de colas
qmgr-name.NO.QSG.CHECKS	No se realizan comprobaciones a nivel de grupo de compartición de colas para este gestor de colas
qsg-name.NO.QSG.CHECKS	No se realizan comprobaciones a nivel de grupo de compartición de colas para este grupo de compartición de colas
qmgr-name.YES.QSG.CHECKS	Alteración temporal de las comprobaciones a nivel de grupo de compartición de colas para este gestor de colas

Si la seguridad de subsistema está activa, no puede desactivar la seguridad a nivel de grupo de compartición de colas y a nivel de gestor de colas. Si intenta hacer esto, IBM MQ activa la comprobación de seguridad en ambos niveles.

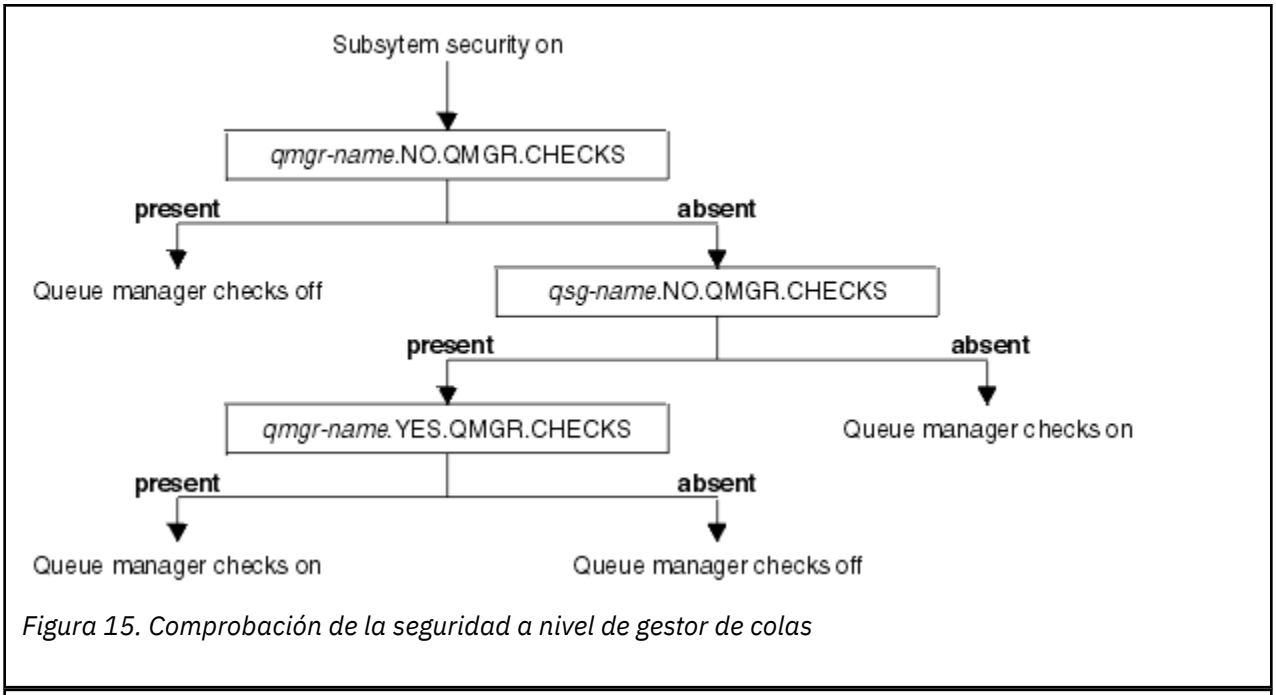


Figura 15. Comprobación de la seguridad a nivel de gestor de colas

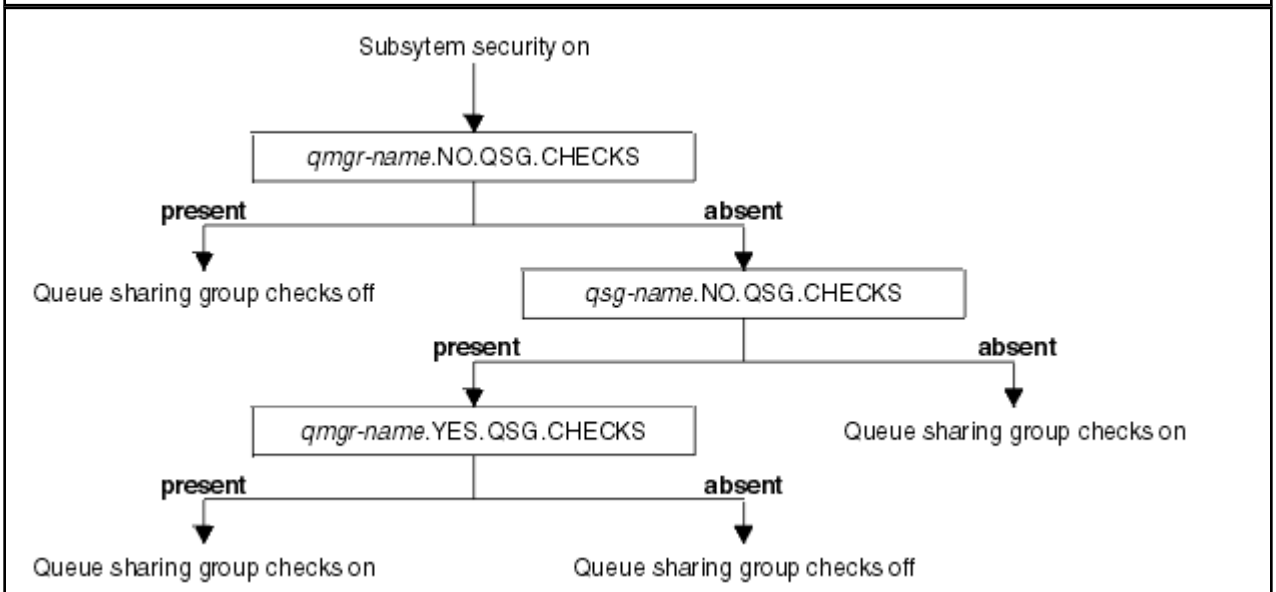


Figura 16. Comprobación de la seguridad a nivel de grupo de compartición de colas

z/OS *Combinaciones válidas de conmutadores de seguridad*

Sólo son válidas ciertas combinaciones de conmutadores. Si utiliza una combinación de valores de conmutador que no es válida, se emite el mensaje CSQH026I y la comprobación de seguridad se activa tanto a nivel de grupo de compartición de colas como a nivel de gestor de colas.

En la Tabla 26 en la página 193, la Tabla 27 en la página 193, la Tabla 28 en la página 193 y la Tabla 29 en la página 194 se muestran los conjuntos de combinaciones de ajustes de conmutador que son válidos para cada tipo de nivel de seguridad.

Tabla 26. Combinaciones válidas de conmutadores de seguridad para la seguridad a nivel de gestor de colas

Combinaciones
qmgr-name.NO.QSG.CHECKS
qsg-name.NO.QSG.CHECKS
qmgr-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS
qsg-name.NO.QSG.CHECKS qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS

Tabla 27. Combinaciones válidas de conmutadores de seguridad para la seguridad a nivel de grupo de compartición de colas

Combinaciones
qmgr-name.NO.QMGR.CHECKS
qsg-name.NO.QMGR.CHECKS
qmgr-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS

Tabla 28. Combinaciones válidas de conmutadores de seguridad para la seguridad a nivel de gestor de colas y de grupo de compartición de colas

Combinaciones
qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS Sin QSG.* perfiles definidos
Sin QMGR.* perfiles definidos qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qmgr-name.YES.QMGR.CHECKS qsg-name.NO.QSG.CHECKS qmgr-name.YES.QSG.CHECKS
Ningún perfil definido para cualquiera de los dos conmutadores

*Tabla 29. Otras combinaciones válidas de conmutadores de seguridad que **activan** ambos niveles de comprobación.*
Combinaciones
qmgr-name.NO.QMGR.CHECKS qmgr-name.NO.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS
qmgr-name.NO.QMGR.CHECKS qsg-name.NO.QSG.CHECKS
qsg-name.NO.QMGR.CHECKS qmgr-name.NO.QSG.CHECKS

z/OS *Comprobaciones a nivel de recurso*

Se utilizan varios perfiles de conmutador para controlar el acceso a los recursos. Algunos impiden que se realicen comprobaciones en un gestor de colas o en un grupo de compartición de colas. Estos perfiles se pueden alterar temporalmente mediante perfiles que permitan la comprobación para gestores de colas específicos.

En Tabla 30 en la página 194 se muestran los perfiles de conmutador para controlar el acceso a los recursos de IBM MQ.

Si el gestor de colas forma parte de un grupo de compartición de colas y tiene activa la seguridad del gestor de colas y del grupo de compartición de colas, puede utilizar un archivo YES.* perfil de conmutación para alterar temporalmente los perfiles de nivel de grupo de compartición de colas y activar específicamente la seguridad para un gestor de colas determinado.

Algunos perfiles se aplican tanto a gestores de colas como a grupos de compartición de colas. Estos perfiles llevan como prefijo la serie *hlq* y debería sustituirla por el nombre de su grupo de compartición de colas o gestor de colas, según sea el caso. Los nombres de perfil que se muestran con el prefijo *qmgr-name* son perfiles de sustitución del gestor de colas; debería sustituir el nombre del gestor de colas.

Tabla 30. Perfiles de conmutador para comprobación de recursos

Tipo de comprobación de recursos que se controla	**Nombre del perfil de conmutador**	**Perfil de alteración temporal para un gestor de colas específico**
Seguridad de conexión	hlq.NO.CONNECT.CHECKS	qmgr-name.YES.CONNECT.CHECKS
Seguridad de colas	hlq.NO.QUEUE.CHECKS	qmgr-name.YES.QUEUE.CHECKS
Seguridad de procesos	hlq.NO.PROCESS.CHECKS	qmgr-name.YES.PROCESS.CHECKS
Seguridad de listas de nombres	hlq.NO.NLIST.CHECKS	qmgr-name.YES.NLIST.CHECKS
Seguridad de contexto	hlq.NO.CONTEXT.CHECKS	qmgr-name.YES.CONTEXT.CHECKS
Seguridad de usuario alternativo	hlq.NO.ALTERNATE.USER.CHECKS	qmgr-name.YES.ALTERNATE.USER.CHECKS

Tabla 30. Perfiles de conmutador para comprobación de recursos (continuación)		
Tipo de comprobación de recursos que se controla	**Nombre del perfil de conmutador**	**Perfil de alteración temporal para un gestor de colas específico**
Seguridad de mandatos	hlq.NO.CMD.CHECKS	qmgr-name.YES.CMD.CHECKS
Seguridad de recursos de mandatos	hlq.NO.CMD.RESC.CHECKS	qmgr-name.YES.CMD.RESC.CHECKS
Seguridad de temas	hlq.NO.TOPIC.CHECKS	qmgr-name.YES.TOPIC.CHECKS
Nota: Perfiles de conmutador genéricos como, por ejemplo, hlq.NO. * * son ignorados por IBM MQ		

Por ejemplo, si desea realizar comprobaciones de seguridad de procesos en el gestor de colas QM01, que es miembro del grupo de compartición de colas QSG3, pero no desea realizar comprobaciones de seguridad de procesos en ninguno de los otros gestores de colas del grupo, defina los siguientes perfiles de conmutador:

```
QSG3.NO.PROCESS.CHECKS
QM01.YES.PROCESS.CHECKS
```

Si desea que se realicen comprobaciones de seguridad de colas en todos los gestores de colas del grupo de compartición de colas, excepto en QM02, defina el siguiente perfil de conmutador:

```
QM02.NO.QUEUE.CHECKS
```

(No hace falta definir un perfil para el grupo de compartición de colas porque las comprobaciones se habilitan automáticamente si no hay ningún perfil definido.)

z/OS *Un ejemplo de cómo definir conmutadores*

Diferentes subsistemas IBM MQ tienen diferentes requisitos de seguridad, que se pueden implementar utilizando diferentes perfiles de conmutador.

Se han definido cuatro subsistemas IBM MQ:

- MQP1 (un sistema de producción)
- MQP2 (un sistema de producción)
- MQD1 (un sistema de desarrollo)
- MQT1 (un sistema de prueba)

Los cuatro gestores de colas son miembros del grupo de compartición de colas QS01. Se han definido y activado todas las clases IBM MQ RACF.

Estos subsistemas tienen requisitos de seguridad diferentes:

- Los sistemas de producción requieren que la comprobación de seguridad completa de IBM MQ esté activa a nivel de grupo de compartición de colas en ambos sistemas.

  Esto se hace especificando el siguiente perfil:

  ```
  RDEFINE MQADMIN QS01.NO.QMGR.CHECKS
  ```

  Esto establece la comprobación a nivel de grupo de compartición de colas para todos los gestores de colas del grupo de compartición de colas. No es necesario definir ningún otro perfil de conmutador para los gestores de colas de producción porque desea comprobarlo todo para estos sistemas.

- El gestor de colas de prueba MQT1 también requiere comprobación de seguridad completa. Sin embargo, puesto que es posible que desee cambiar esto más tarde, la seguridad se puede definir en el nivel del gestor de colas, de modo que pueda cambiar los valores de seguridad para este gestor de colas sin afectar a los demás miembros del grupo de compartición de colas.

  Esto se hace definiendo el perfil NO.QSG.CHECKS para MQT1 como se indica a continuación:

  ```
  RDEFINE MQADMIN MQT1.NO.QSG.CHECKS
  ```

- El gestor de colas de desarrollo MQD1 tiene requisitos de seguridad diferentes de los del resto del grupo de compartición de colas. Requiere que sólo la seguridad de conexión y de colas esté activa.

  Esto se hace definiendo un perfil `MQD1.YES.QMGR.CHECKS` para este gestor de colas, y luego definiendo los siguientes perfiles para desactivar la comprobación de seguridad para los recursos que no necesitan comprobarse:

  ```
  RDEFINE MQADMIN MQD1.NO.CMD.CHECKS
  RDEFINE MQADMIN MQD1.NO.CMD.RESC.CHECKS
  RDEFINE MQADMIN MQD1.NO.PROCESS.CHECKS
  RDEFINE MQADMIN MQD1.NO.NLIST.CHECKS
  RDEFINE MQADMIN MQD1.NO.CONTEXT.CHECKS
  RDEFINE MQADMIN MQD1.NO.ALTERNATE.USER.CHECKS
  ```

Cuando el gestor de colas está activo, puede visualizar los valores de seguridad actuales emitiendo el mandato MQSC DISPLAY SECURITY.

También puede cambiar los valores de conmutador cuando el gestor de colas está en ejecución, definiendo o suprimiendo el perfil de conmutador adecuado en la clase MQADMIN. Para activar los cambios realizados en los valores de conmutador, debe emitir el mandato REFRESH SECURITY para la clase MQADMIN.

Consulte “Renovación de la seguridad del gestor de colas en z/OS” en la página 252 para obtener información detallada sobre el uso de los mandatos DISPLAY SECURITY y REFRESH SECURITY.

z/OS Perfiles utilizados para controlar el acceso a los recursos de IBM MQ

Debe definir perfiles RACF para controlar el acceso a los recursos de IBM MQ, además de los perfiles de conmutador que puedan haberse definido. Esta colección de temas contiene información sobre los perfiles RACF para los diferentes tipos de recursos de IBM MQ.

Si no tiene un perfil de recurso definido para una comprobación de seguridad específica, y un usuario emite una solicitud que implicaría la realización de esa comprobación, IBM MQ deniega el acceso. No es necesario que defina perfiles para los tipos de seguridad relativos a cualquier conmutador de seguridad que haya desactivado.

z/OS *Perfiles para la seguridad de conexión*

Si la seguridad de conexión está activa, debe definir perfiles en la clase MQCONN y permitir que los grupos o los ID de usuario necesarios accedan a dichos perfiles, para que puedan conectarse a IBM MQ.

Para permitir establecer una conexión, debe otorgar a los usuarios acceso READ de RACF al perfil adecuado. (Si no existe ningún perfil a nivel de gestor de colas, y el gestor de colas es miembro de un grupo de compartición de colas, pueden realizarse comprobaciones en perfiles a nivel de grupo de compartición de colas, si la seguridad está configurada para hacerlo).

Un perfil de conexión calificado con un nombre de gestor de colas controla el acceso a un gestor de colas específico, y los usuarios a los que se ha otorgado acceso a este perfil puede conectarse a dicho gestor de colas. Un perfil de conexión calificado con un nombre de grupo de compartición de colas controla el acceso a todos los gestores de colas del grupo de compartición de colas para dicho tipo de conexión. Por ejemplo, un usuario con acceso a `QS01.BATCH` puede utilizar una conexión por lotes a cualquier gestor

de colas del grupo de compartición de colas QS01 que no tenga definido un perfil a nivel de gestor de colas.

Nota:

1. Para obtener información sobre los ID de usuario que se comprueban para diferentes peticiones de seguridad, consulte “ID de usuario para la comprobación de seguridad en z/OS” en la página 240.
2. También se realizan comprobaciones de seguridad a nivel de recurso (RESLEVEL) durante la conexión. Para obtener detalles, consulte “El perfil de seguridad RESLEVEL” en la página 233.

La seguridad de IBM MQ reconoce los siguientes tipos de conexión:

- Conexiones por lotes (y de tipo por lotes), que incluyen:
  - Trabajos por lotes de z/OS
  - Aplicaciones TSO
  - Inicios de sesión USS
  - Procedimientos almacenados de Db2
- Conexiones CICS
- Conexiones IMS desde regiones de control y de proceso de aplicaciones
- El iniciador de canal de IBM MQ

z/OS *Perfiles de seguridad de conexión para conexiones por lotes*

Los perfiles para comprobar las conexiones de tipo por lotes están formados por el nombre del gestor de colas o del grupo de compartición de colas seguido de la palabra *BATCH*. Otorgue al ID de usuario asociado al espacio de direcciones de conexión acceso de lectura (READ) al perfil de conexión.

Los perfiles para comprobar las conexiones por lotes y de tipo por lotes adoptan la forma:

```
hlq.BATCH
```

donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas). Si está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas. Si no encuentra ninguno de los dos perfiles, la solicitud de conexión no se realiza correctamente.

Para las solicitudes de conexión por lotes o de tipo por lotes, debe permitir que el ID de usuario asociado al espacio de direcciones de conexión acceda el perfil de conexión. Por ejemplo, el siguiente mandato RACF permite a los usuarios del grupo CONNTQM1 conectarse al gestor de colas TQM1; a estos ID de usuario se les permitirá utilizar cualquier conexión por lotes o de tipo por lotes.

```
RDEFINE MQCONN TQM1.BATCH UACC(NONE)
PERMIT TQM1.BATCH CLASS(MQCONN) ID(CONNTQM1) ACCESS(READ)
```

z/OS *Utilización de **CHCKLOCL** en aplicaciones enlazadas localmente*

CHCKLOCL solamente se aplica a conexiones que se realizan a través de conexiones BATCH y no se aplican a conexiones realizadas desde CICS o IMS. Las conexiones realizadas a través del iniciador de canal se controlan mediante **CHCKCLNT**.

Visión general

Si desea configurar el gestor de colas de z/OS de modo que requiera la comprobación de ID de usuario y contraseña para algunas, pero no todas, las aplicaciones enlazadas localmente, es necesario realizan alguna configuración adicional.

La razón para ello es que una vez que se configura **CHCKLOCL** (*REQUIRED*), las aplicaciones por lotes heredadas que utilizan la llamada de API MQCONN ya no podrán conectarse al gestor de colas.

Solo para z/OS, se puede utilizar un mecanismo más granular basado en la seguridad de la conexión de un espacio de direcciones para revertir la configuración global de CHCKLOCL(REQUIRED) a CHCKLOCL(OPTIONAL) para los ID de usuarios definidos de forma específica. El mecanismo utilizado se describe en el siguiente texto, junto con un ejemplo.

Para permitir más granularidad en **CHCKLOCL** (*REQUIRED*) que solo EVERYONE, puede modificar **CHCKLOCL** de la misma forma que modifica el nivel de acceso del ID de usuario asociado con el espacio de direcciones de conexión para los perfiles de conexión `hlq.batch` en la clase MQCONN.

Si el ID de usuario de espacio de direcciones solo tiene acceso READ, que es el mínimo necesario para poder conectarse, la configuración de **CHCKLOCL** se aplica tal como se escribe.

Si el ID de usuario del espacio de direcciones tiene acceso UPDATE (o superior), la configuración **CHCKLOCL** opera en modalidad *OPTIONAL*. Es decir, no tiene que proporcionar un ID de usuario y una contraseña, pero si lo hace, el ID de usuario y la contraseña deben ser un par válido.

La conexión de seguridad ya está configurada para el gestor de colas de z/OS

Si tiene configurada la seguridad de conexión para el gestor de colas z/OS y desea que **CHCKLOCL** (*REQUIRED*) se aplique a aplicaciones WAS enlazadas localmente y no a otras, lleve a cabo los siguientes pasos:

1. Empiece con **CHCKLOCL** (*OPTIONAL*) como su configuración. Esto significa que se comprueba la validez de todos los ID de usuario y todas las contraseñas que se proporcionen, pero no es obligatorio.
2. Liste todos los usuarios que tienen acceso a los perfiles de seguridad de conexiones mediante el mandato:

   ```
   RLIST MQCONN MQ23.BATCH AUTHUSER
   ```

   Este mandato muestra, por ejemplo:

   ```
   CLASS    NAME
   -----    ----
   MQCONN   MQ23.BATCH

   USER     ACCESS  ACCESS COUNT
   ----     ------  ------ -----
   JOHNDOE  READ    000009
   JDOE1    READ    000003
   WASUSER  READ    000000
   ```

3. Para cada ID de usuario listado para tener acceso READ, cambie el acceso por

   ```
   UPDATE:- PERMIT MQ23.BATCH CLASS(MQCONN) ID(JOHNDOE) ACCESS(UPDATE)
   ```

4. Actualice la configuración IBM MQ a **CHCKLOCL** (*REQUIRED*).

   La combinación del acceso UPDATE a `MQ23.BATCH` y el valor actual significa que está utilizando **CHCKLOCL** (*OPTIONAL*).
5. Ahora, aplique el comportamiento **CHCKLOCL** (*REQUIRED*) a un ID de usuario específico, por ejemplo, WASUSER, de modo que todas las conexiones procedentes de esa región deben proporcionar un ID de usuario y una contraseña.

   Hágalo invirtiendo el cambio que ha realizado anteriormente emitiendo el mandato:

   ```
   PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
   ```

La conexión de seguridad no está configurada para el gestor de colas de z/OS

En esta situación, debe:

1. Crear perfiles de conexión para `hlq.BATCH` en la clase MQCONN, emitiendo el mandato:

   ```
   RDEFINE MQCONN MQ23.BATCH UACC(NONE)
   ```

2. Autorizar todos los ID de usuarios que crean conexiones por lotes con el gestor de colas para que tengan acceso UPDATE a este perfil. Al hacerlo pasa por alto el requisito **CHCKLOCL** (*REQUIRED*) para el ID de usuario y la contraseña en el momento de la conexión.

   Hágalo emitiendo el mandato:

   ```
   PERMIT MQ23.BATCH CLASS(MQCONN)ID(JOHNDOE) ACCESS(UPDATE)
   ```

   Estos incluyen los ID de usuario:

   a. Utilizados para CSQUTIL, paneles ISPF y otras herramientas enlazadas localmente.

   b. Asociados con conexiones como lotes con el gestor de colas. Por ejemplo, piense en procedimientos almacenados de Advanced Message Security, IBM Integration Bus, Db2 usuarios USS y TSO, y aplicaciones Java

3. Suprimir el perfil de conmutación para el gestor de colas emitiendo el mandato:

   ```
   hlq.NO.CONNECT.CHECKS
   ```

4. Ahora, aplique el comportamiento **CHCKLOCL** (*REQUIRED*) a un ID de usuario específico, por ejemplo, WASUSER, de modo que todas las conexiones procedentes de esa región deben proporcionar un ID de usuario y una contraseña.

   Hágalo invirtiendo el cambio que ha realizado anteriormente emitiendo el mandato:

   ```
   PERMIT MQ23.BATCH CLASS(MQCONN) ID(WASUSER) ACCESS(READ)
   ```

z/OS *Perfiles de seguridad de conexión para conexiones CICS*

Los perfiles para comprobar conexiones CICS están formados por el nombre del gestor de colas o del grupo de compartición de colas seguido de la palabra *CICS*. Otorgue al ID de usuario asociado al espacio de direcciones de CICS acceso de lectura (READ) al perfil de conexión.

Los perfiles para comprobar las conexiones del CICS adoptan la forma:

```
hlq.CICS
```

donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas). Si está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas. Si no encuentra ninguno de los dos perfiles, la solicitud de conexión no se realiza correctamente.

Para las solicitudes de conexión mediante CICS, solo tiene que permitir el ID de usuario del espacio de direcciones de CICS el acceso al perfil de conexión.

Por ejemplo, los mandatos RACF siguientes permiten al ID de usuario del espacio de direcciones CICS KCBCICS conectarse al gestor de colas TQM1:

```
RDEFINE MQCONN TQM1.CICS UACC(NONE)
PERMIT TQM1.CICS CLASS(MQCONN) ID(KCBCICS) ACCESS(READ)
```

z/OS *Perfiles de seguridad de conexión para conexiones IMS*

Los perfiles para comprobar conexiones IMS están formados por el nombre del gestor de colas o del grupo de compartición de colas seguido de la palabra *IMS*. Otorgue a los ID de usuario de las regiones dependiente y de control de IMS acceso de lectura (READ) al perfil de conexión.

Los perfiles para comprobar las conexiones del IMS adoptan la forma:

```
hlq.IMS
```

donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas). Si está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas. Si no encuentra ninguno de los dos perfiles, la solicitud de conexión no se realiza correctamente.

Para las solicitudes de conexión mediante IMS, permita el acceso al perfil de conexión para los ID de usuario de las regiones dependiente y de control de IMS.

Por ejemplo, el siguiente mandato RACF permite que:

- El ID de usuario de región de IMS, IMSREG, se conecte al gestor de colas TQM1.
- Los usuarios del grupo BMPGRP sometan trabajos BMP.

```
RDEFINE MQCONN TQM1.IMS UACC(NONE)
PERMIT TQM1.IMS CLASS(MQCONN) ID(IMSREG,BMPGRP) ACCESS(READ)
```

z/OS *Perfiles de seguridad de conexión para el iniciador de canal*

Los perfiles para comprobar las conexiones del iniciador de canal están formados por el nombre del gestor de colas o del grupo de compartición de colas, seguido de la palabra *CHIN*. Otorgue al ID de usuario utilizado por el espacio de direcciones de tareas iniciadas del iniciador de canal acceso de lectura (READ) al perfil de conexión.

Los perfiles para comprobar las conexiones del iniciador de canal adoptan la forma:

```
hlq.CHIN
```

donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas). Si está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas. Si no encuentra ninguno de los dos perfiles, la solicitud de conexión no se realiza correctamente.

Para las solicitudes de conexión realizadas por el iniciador de canal, defina el acceso al perfil de conexión para el ID de usuario utilizado por el espacio de direcciones de tareas iniciadas del iniciador de canal.

Por ejemplo, los siguientes mandatos RACF permiten al espacio de direcciones del iniciador de canal que se ejecuta con el ID de usuario DQCTRL conectarse al gestor de colas TQM1:

```
RDEFINE MQCONN TQM1.CHIN UACC(NONE)
PERMIT TQM1.CHIN CLASS(MQCONN) ID(DQCTRL) ACCESS(READ)
```

z/OS *Perfiles para la seguridad de colas*

Si la seguridad de colas está activa, debe definir perfiles en las clases adecuadas y permitir a los grupos o los ID de usuario necesarios acceder a dichos perfiles. Se nombran los perfiles de seguridad de colas después del gestor de colas o del grupo de compartición de colas y la cola que se va a abrir.

Si la seguridad de colas está activa, debe:

- Definir perfiles en las clases **MQQUEUE** o **GMQQUEUE** si se utilizan perfiles en mayúsculas.
- Definir perfiles en las clases **MXQUEUE** o **GMXQUEUE** si se utilizan perfiles que combinan mayúsculas y minúsculas.
- Permitir a los grupos o los ID de usuario necesarios acceder a estos perfiles, para que puedan emitir peticiones API de IBM MQ que utilicen colas.

Los perfiles para la seguridad de colas adoptan la forma:

```
hlq.queuename
```

donde `hlq` puede ser `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas), y `queuename` es el nombre de la cola que se abre, tal como se especifica en el descriptor de objetos de la llamada `MQOPEN` o `MQPUT1`.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el acceso a una única cola de dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el acceso a una o más colas con dicho nombre de cola en todos los gestores de colas del grupo de compartición de colas, o el acceso a una cola compartida por cualquier gestor de colas del grupo. Este acceso se puede modificar en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para esa cola en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

Si está utilizando colas compartidas, se recomienda que utilice la seguridad a nivel de grupo de compartición de colas.

Para obtener información detallada de cómo funciona la seguridad de colas cuando el nombre de cola es el de una cola alias o modelo z/OS, consulte “Consideraciones para las colas alias” en la página 203 y “Consideraciones para las colas modelo” en la página 204 .

El acceso RACF que se necesita para abrir una cola depende de las opciones `MQOPEN` o `MQPUT1` que se especifiquen. Si se codifica más de una de las opciones MQOO_* y MQPMO_*, la comprobación de seguridad de colas se realiza para la autorización RACF más alta requerida.

Tabla 31. Niveles de acceso para seguridad de colas utilizando las llamadas MQOPEN o MQPUT1

Opción MQOPEN o MQPUT1	**Nivel de acceso de RACF necesario para hlq.queuename**
MQOO_BROWSE	READ
MQOO_INQUIRE	READ
MQOO_BIND_*	UPDATE
MQOO_INPUT_*	UPDATE
MQOO_OUTPUT o MQPUT1	UPDATE
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	UPDATE

Tabla 31. Niveles de acceso para seguridad de colas utilizando las llamadas MQOPEN o MQPUT1 (continuación)

Opción MQOPEN o MQPUT1	**Nivel de acceso de RACF necesario para hlq.queuename**
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	UPDATE
MQOO_SAVE_ALL_CONTEXT	UPDATE
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	UPDATE
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	UPDATE
MQOO_SET	ALTER

Por ejemplo, en el gestor de colas de IBM MQ QM77, a todos los ID de usuario del grupo de RACF PAYGRP se les debe otorgar acceso para obtener mensajes de o colocar mensajes en todas las colas con nombres que empiecen por 'PAY.'. Puede hacerlo utilizando estos mandatos RACF:

```
RDEFINE MQQUEUE QM77.PAY.** UACC(NONE)
PERMIT QM77.PAY.** CLASS(MQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

Además, todos los ID de usuario del grupo PAYGRP deben tener acceso para transferir mensajes a colas que no sigan el convenio de denominación PAY. Por ejemplo:

```
REQUEST_QUEUE_FOR_PAYROLL
SALARY.INCREASE.SERVER
REPLIES.FROM.SALARY.MODEL
```

Puede hacer esto definiendo perfiles para estas colas en la clase GMQQUEUE y otorgando acceso a esa clase como se indica a continuación:

```
RDEFINE GMQQUEUE PAYROLL.EXTRAS UACC(NONE)
        ADDMEM(QM77.REQUEST_QUEUE_FOR_PAYROLL,
               QM77.SALARY.INCREASE.SERVER,
               QM77.REPLIES.FROM.SALARY.MODEL)
PERMIT PAYROLL.EXTRAS CLASS(GMQQUEUE) ID(PAYGRP) ACCESS(UPDATE)
```

Nota:

1. Si se cambia el nivel de acceso RACF que una aplicación tiene a un perfil de seguridad de colas, los cambios sólo tienen efecto para los nuevos manejadores de objeto obtenidos (es decir, nuevas MQOPEN) para esa cola. Los manejadores que ya existían en el momento del cambio mantienen su acceso existente a la cola. Si una aplicación necesita utilizar el nivel de acceso cambiado a la cola, en lugar del nivel de acceso existente, ésta debe cerrar y volver a abrir la cola para cada manejador de objeto que requiera el cambio.
2. En el ejemplo, el nombre de gestor de colas QM77 podría ser también el nombre de un grupo de compartición de colas.

Es posible que también se produzcan otros tipos de comprobaciones de seguridad cuando se abra la cola, dependiendo de las opciones de apertura especificadas y de los tipos de seguridad que estén activos.

z/OS Consulte también “Perfiles para la seguridad de contexto” en la página 218 y “Perfiles para

la seguridad de usuario alternativo" en la página 216. Para ver una tabla de resumen que muestra las opciones de apertura y las autorizaciones de seguridad necesarias cuando la seguridad de colas, de contexto y de usuario alternativo están todas activas, consulte Tabla 36 en la página 209.

Si utiliza la publicación/suscripción, debe tener en cuenta lo siguiente. Cuando se procesa una solicitud MQSUB, se realiza una comprobación de seguridad para asegurar que el ID de usuario que realiza la solicitud tiene el acceso necesario para transferir mensajes a la cola IBM MQ de destino, así como el acceso necesario para suscribirse al tema de IBM MQ.

Tabla 32. Niveles de acceso para seguridad de colas utilizando la llamada MQSUB

Opción MQSUB	**Nivel de acceso de RACF necesario para hlq.queuename**
MQSO_ALTER, MQSO_CREATE y MQSO_RESUME	UPDATE

Nota:

1. `hlq.queuename` es la cola de destino para las publicaciones. Cuando ésta es una cola gestionada, necesita que se utilice el acceso a la cola modelo adecuada para la cola gestionada y la cola dinámica que se crean.
2. Puede utilizar una técnica como ésta para la cola de destino que proporcione en una llamada API MQSUB, si desea distinguir entre los usuarios que realizan las suscripciones y los usuarios que recuperan las publicaciones de la cola de destino.

z/OS *Consideraciones para las colas alias*

Cuando se emite una llamada MQOPEN o MQPUT1 para una cola alias, IBM MQ realiza una comprobación de recursos para el nombre de cola especificado en el descriptor de objeto (MQOD) de la llamada. No comprueba si el usuario tiene permiso para acceder al nombre de la cola de destino.

Por ejemplo, una cola alias llamada PAYROLL.REQUEST se resuelve en una cola de destino PAY.REQUEST. Si la seguridad de colas está activa, sólo necesita tener autorización para acceder a la cola PAYROLL.REQUEST. No se realiza ninguna comprobación para ver si tiene autorización para acceder a la cola PAY.REQUEST.

z/OS *Utilización de colas alias para distinguir entre peticiones MQGET y MQPUT*

El rango de llamadas MQI disponibles en un nivel de acceso puede ocasionar un problema si desea restringir el acceso a una cola para permitir sólo la llamada **MQPUT** o sólo la llamada **MQGET**. Una cola se puede proteger mediante la definición de dos alias que se resuelvan en esa cola: uno que permita a las aplicaciones obtener mensajes de la cola y otro que permita a las aplicaciones transferir mensajes a la cola.

El texto siguiente le ofrece un ejemplo de cómo puede definir las colas en IBM MQ:

```
DEFINE QLOCAL(MUST_USE_ALIAS_TO_ACCESS) GET(ENABLED)
       PUT(ENABLED)

DEFINE QALIAS(USE_THIS_ONE_FOR_GETS) GET(ENABLED)
       PUT(DISABLED) TARGET(MUST_USE_ALIAS_TO_ACCESS)

DEFINE QALIAS(USE_THIS_ONE_FOR_PUTS) GET(DISABLED)
       PUT(ENABLED) TARGET(MUST_USE_ALIAS_TO_ACCESS)
```

También debe crear las siguientes definiciones RACF:

```
RDEFINE MQQUEUE hlq.MUST_USE_ALIAS_TO_ACCESS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_GETS UACC(NONE)
RDEFINE MQQUEUE hlq.USE_THIS_ONE_FOR_PUTS UACC(NONE)
```

A continuación, asegúrese de que ningún usuario tenga acceso a la cola hlq.MUST_USE_ALIAS_TO_ACCESS, y otorgue a los usuarios o grupos adecuados acceso a la cola alias. Puede hacerlo utilizando estos mandatos RACF:

```
PERMIT hlq.USE_THIS_ONE_FOR_GETS CLASS(MQQUEUE)
       ID(GETUSER,GETGRP) ACCESS(UPDATE)
PERMIT hlq.USE_THIS_ONE_FOR_PUTS CLASS(MQQUEUE)
       ID(PUTUSER,PUTGRP) ACCESS(UPDATE)
```

Esto significa que al ID de usuario GETUSER y a los ID de usuario del grupo GETGRP sólo se les permite obtener mensajes de MUST_USE_ALIAS_TO_ACCESS mediante la cola alias USE_THIS_ONE_FOR_GETS; y al ID de usuario PUTUSER y a los ID de usuario del grupo PUTGRP sólo se les permite transferir mensajes mediante la cola alias USE_THIS_ONE_FOR_PUTS.

Nota:

1. Si desea utilizar una técnica como ésta, debe informar a los desarrolladores de aplicaciones, para que puedan diseñar adecuadamente los programas.
2. Puede utilizar una técnica como esta para la cola de destino que proporciona en una solicitud de API MQSUB si desea distinguir entre los usuarios que realizan las suscripciones y los usuarios que 'obtienen' las publicaciones de la cola de destino.

z/OS *Consideraciones para las colas modelo*

Para abrir una cola modelo, debe poder abrir la propia cola modelo y la cola dinámica en la que se resuelve. Defina perfiles RACF genéricos para las colas dinámicas, incluyendo las colas dinámicas que utilizan los programas de utilidad de IBM MQ.

Cuando se abre una cola modelo, la seguridad de IBM MQ realiza dos comprobaciones de seguridad de colas:

1. ¿Tiene autorización para acceder a la cola modelo?
2. ¿Tiene autorización para acceder a la cola dinámica en la que se resuelve la cola modelo?

Si el nombre de la cola dinámica contiene un carácter asterisco (*) final, este carácter * se sustituye por una serie de caracteres generada por IBM MQ, para crear una cola dinámica con un nombre exclusivo. Sin embargo, puesto que el nombre completo, incluyendo esta serie generada, se utiliza para comprobar la autorización, debería definir perfiles genéricos para estas colas.

Por ejemplo, una llamada MQOPEN utiliza un nombre de cola modelo de CREDIT.CHECK.REPLY.MODEL y un nombre de cola dinámica de CREDIT.REPLY.* en el gestor de colas (o grupo de compartición de colas) MQSP.

Para hacer esto, debe emitir los siguientes mandatos RACF para definir los perfiles de cola necesarios:

```
RDEFINE MQQUEUE MQSP.CREDIT.CHECK.REPLY.MODEL
RDEFINE MQQUEUE MQSP.CREDIT.REPLY.**
```

También debe emitir los mandatos PERMIT de RACF correspondientes para permitir al usuario acceso a estos perfiles.

Un nombre típico de cola dinámica creado por una llamada MQOPEN es algo parecido a CREDIT.REPLY.A346EF00367849A0. El valor preciso del último calificador es imprevisible; esta es la razón por la que debería utilizar perfiles genéricos para estos nombres de cola.

Hay una serie de programas de utilidad de IBM MQ que transfieren mensajes a colas dinámicas. Debería definir perfiles para los siguientes nombres de colas dinámicas, y proporcionar acceso RACF UPDATE a los ID de usuario adecuados (consulte “ID de usuario para la comprobación de seguridad en z/OS” en la página 240 para ver los ID de usuario correctos):

```
SYSTEM.CSQUTIL.*  (used by CSQUTIL)
SYSTEM.CSQOREXX.* (used by the operations and control panels)
SYSTEM.CSQXCMD.*  (used by the channel initiator when processing CSQINPX)
CSQ4SAMP.*        (used by the IBM MQ supplied samples)
```

También podría considerar la posibilidad de definir un perfil para controlar el uso del nombre de cola dinámica utilizado de forma predeterminada en los miembros de copia de programación de aplicaciones. Los libros de copias proporcionados por IBM MQcontienen un *DynamicQName*predeterminado, que es CSQ. *. Esto permite establecer un perfil RACF adecuado.

Nota: No permita a los programadores de aplicaciones especificar un solo * para el nombre de cola dinámica. Si lo hace, debe definir un hlq. * * en la clase MQQUEUE, y tendría que darle un acceso de gran alcance. Esto significa que este perfil podría utilizarse también para otras colas no dinámicas que no tengan un perfil RACF más específico. Los usuarios podrían, por tanto, obtener acceso a colas a las que no desea que accedan.

z/OS *Opciones de cierre en colas dinámicas permanentes*

Si una aplicación abre una cola dinámica permanente que ha sido creada por otra aplicación y luego intenta suprimir esa cola con una opción MQCLOSE, se aplican algunas comprobaciones de seguridad adicionales cuando se realiza el intento.

Tabla 33. Niveles de acceso para opciones de cierre en colas dinámicas permanentes

Opción MQCLOSE	**Nivel de acceso de RACF necesario para hlq.queuename**
MQCO_DELETE	ALTER
MQCO_DELETE_PURGE	ALTER

z/OS *Seguridad y colas remotas*

Cuando un mensaje se transfiere a una cola remota, la seguridad de colas que implementa el gestor de colas local depende de cómo se especifica la cola remota cuando se abre.

Se aplican las siguientes reglas:

1. Si la cola remota se ha definido en el gestor de colas local mediante el mandato IBM MQ DEFINE QREMOTE, la cola que se comprueba es el nombre de la cola remota. Por ejemplo, si se define una cola remota en el gestor de colas MQS1 de la siguiente manera:

   ```
   DEFINE QREMOTE(BANK7.CREDIT.REFERENCE)
          RNAME(CREDIT.SCORING.REQUEST)
          RQMNAME(BNK7)
          XMITQ(BANK1.TO.BANK7)
   ```

   En este caso, se debe definir un perfil BANK7.CREDIT.REFERENCE en la clase MQQUEUE.

2. Si el *NombGstColasObj* de la solicitud no se resuelve en el gestor de colas local, se lleva a cabo una comprobación de seguridad en el nombre de gestor de colas (remoto) resuelto, salvo en el caso de una cola de clúster, donde la comprobación se realiza en el nombre de cola de clúster.

   Por ejemplo, la cola de transmisión BANK1.TO.BANK7 se define en el gestor de colas MQS1. A continuación se emite una solicitud MQPUT1 en MQS1 que especifica *NombreObjeto* como BANK1.INTERBANK.TRANSFERS y un *NombGstColasObj* de BANK1.TO.BANK7. En este caso, el usuario que realiza la solicitud debe tener acceso a BANK1.TO.BANK7.

3. Si realiza una solicitud MQPUT a una cola y especifica *ObjectQMgrName* como el nombre de un alias del gestor de colas local, sólo se comprueba la seguridad del nombre de cola, no la del gestor de colas.

Cuando el mensaje llega al gestor de colas remoto, puede ser sometido a un proceso de seguridad adicional. Para obtener más información, consulte “Seguridad de la mensajería remota” en la página 96.

z/OS *Seguridad de la cola de mensajes no entregados*

Se aplican consideraciones especiales a la cola de mensajes no entregados ya que muchos usuarios deben poder transferir mensajes a la cola, pero el acceso para recuperar mensajes debe estar estrictamente restringido. Para lograr esto, puede aplicar autorizaciones RACF diferentes a la cola de mensajes no entregados y a una cola alias.

Los mensajes no entregados pueden transferirse a una cola especial llamada la cola de mensajes no entregados. Si tiene información confidencial que posiblemente podría acabar en esta cola, debe tener en cuenta las implicaciones de seguridad de este hecho, ya que no desea que usuarios no autorizados recuperen estos datos.

Cada uno de los siguientes elementos debe estar autorizado a transferir mensajes a la cola de mensajes no entregados:

- Programas de aplicación.
- El espacio de direcciones del iniciador de canal y cualquier ID de usuario de MCA. (Si el perfil RESLEVEL no está presente, o está definido para que se comprueben los ID de usuario del canal, el ID de usuario de canal también necesita autorización para transferir mensajes a la cola de mensajes no entregados.)
- CKTI, el iniciador de tareas de CICS proporcionado por CICS.
- CSQQTRMN, el supervisor desencadenante de IBM MQ proporcionado por IMS.

La única aplicación que pueda recuperar mensajes de la cola de mensajes no entregados debería ser una aplicación 'especial' que procese estos mensajes. No obstante, surge un problema si otorga a las aplicaciones autorización RACF UPDATE a la cola de mensajes no entregados para MQPUTs ya que éstas pueden recuperar automáticamente mensajes de la cola mediante llamadas MQGET. No puede inhabilitar la cola de mensajes no entregados para operaciones de obtener porque, si lo hace, ni siquiera las aplicaciones 'especiales' podrían recuperar los mensajes.

Una solución a este problema es configurar un acceso de dos niveles a la cola de mensajes no entregados. CKTI, las transacciones de agente de canal de mensajes o el espacio de direcciones del iniciador de canal, y las aplicaciones 'especiales' tienen acceso directo; otras aplicaciones sólo pueden acceder a la cola de mensajes no entregados a través de una cola alias. Este alias se define para permitir a las aplicaciones transferir mensajes a la cola de mensajes no entregados, pero no para obtener mensajes de la misma.

Así es como podría funcionar:

1. Defina la cola de mensajes no entregados real con los atributos PUT(ENABLED) y GET(ENABLED), tal como se muestra en el ejemplo thlqual.SCSQPROC(CSQ4INYG).
2. Otorgue autorización RACF UPDATE para la cola de mensajes no entregados a los ID de usuario siguientes:
   - Los ID de usuario bajo los que se ejecutan el CKTI y los MCA o el espacio de direcciones del iniciador de canal.
   - Los ID de usuario asociados a la aplicación de proceso 'especial' de la cola de mensajes no entregados.
3. Defina una cola alias que se resuelva en la cola de mensajes no entregados real, pero asigne a la cola alias estos atributos: PUT(ENABLED) y GET(DISABLED). Asigne a la cola alias un nombre con la misma raíz que el nombre de la cola de mensajes no entregados, pero añada los caracteres ".PUT" al final de esta raíz. Por ejemplo, si el nombre de la cola de mensajes no entregados es hlq.DEAD.QUEUE, el nombre de la cola alias será hlq.DEAD.QUEUE.PUT.
4. Para transferir un mensaje a la cola de mensajes no entregados, una aplicación utiliza la cola alias. Esto es lo que la aplicación debe hacer:
   - Recuperar el nombre de la cola de mensajes no entregados real. Para ello, abre el objeto de gestor de colas utilizando MQOPEN y luego emite una llamada MQINQ para obtener el nombre de la cola de mensajes no entregados.
   - Crear el nombre de la cola alias, añadiendo los caracteres '.PUT' a este nombre, en este caso hlq.DEAD.QUEUE.PUT.

- Abrir la cola alias, hlq.DEAD.QUEUE.PUT.
- Transferir el mensaje a la cola de mensajes no entregados real, emitiendo una llamada MQPUT para la cola alias.

5. Otorgue al ID de usuario asociado a la aplicación autorización RACF UPDATE al alias, pero sin acceso (autorización NONE) a la cola de mensajes no entregados real. Esto significa que:
   - La aplicación puede transferir mensajes a la cola de mensajes no entregados mediante la cola alias.
   - La aplicación no puede obtener mensajes de la cola de mensajes no entregados utilizando la cola alias porque ésta está inhabilitada para operación de obtener.

   La aplicación no puede obtener tampoco mensajes de la cola de mensajes no entregados real porque no tiene la autorización RACF correcta.

En la Tabla 34 en la página 207 se resume la autorización RACF necesaria para los diversos participantes en esta solución.

Tabla 34. Autorización RACF para la cola de mensajes no entregados y su alias

Identificadores (ID) de usuario asociados	**Cola de mensajes no entregados real (hlq.DEAD.QUEUE)**	**Cola de mensajes no entregados alias (hlq.DEAD.QUEUE.PUT)**
MCA o espacio de direcciones del iniciador de canal y CKTI	UPDATE	NINGUNO
Aplicación 'especial' (para proceso de la cola de mensajes no entregados)	UPDATE	NINGUNO
Identificadores (ID) de usuario de aplicaciones escritas por el usuario	NINGUNO	UPDATE

Si utiliza este método, la aplicación no puede determinar la longitud máxima de mensaje (MAXMSGL) de la cola de mensajes no entregados. Esto es debido a que el atributo MAXMSGL no se puede recuperar de una cola alias. Por lo tanto, la aplicación debería presuponer que la longitud máxima de mensaje es 100 MB, el tamaño máximo aceptado por IBM MQ for z/OS. La cola de mensajes no entregados real también debería definirse con un atributo MAXMSGL de 100 MB.

Nota: Los programas de aplicación escritos por el usuario normalmente no utilizan autorización de usuario alternativo para transferir mensajes a la cola de mensajes no entregados. Esto reduce el número de identificadores (ID) de usuario que tienen acceso a la cola de mensajes no entregados.

z/OS *Seguridad de colas del sistema*

Debe configurar el acceso RACF para permitir a determinados ID de usuario el acceso a colas del sistema específicas.

A muchas de las colas del sistema acceden los componentes auxiliares de IBM MQ:

- El programa de utilidad CSQUTIL
- El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL)
- Los paneles de operaciones y los paneles de control
- El espacio de direcciones del iniciador de canal (incluido el daemon de publicación/suscripción en cola)
- V 9.1.0 El servidor mqweb, utilizado por MQ Console y REST API.

Los ID de usuario bajo los que se ejecutan estos componentes deben tener acceso RACF a estas colas, tal como se muestra en la Tabla 35 en la página 208.

Tabla 35. Acceso necesario a las colas SYSTEM por parte de IBM MQ

Cola SYSTEM	CSQUTIL	CSQ0UTIL	Servidor mqweb	Paneles de operaciones y paneles de control	Iniciador de canal para gestión de colas distribuidas
SYSTEM.ADMIN.CHANNEL.EVENT	-	-	-	-	UPDATE
SYSTEM.ADMIN.COMMAND.QUEUE	-	-	UPDATE	-	-
SYSTEM.BROKER.ADMIN.STREAM	-	-	-	-	ALTER
SYSTEM.BROKER.CONTROL.QUEUE	-	-	-	-	ALTER
SYSTEM.BROKER.DEFAULT.STREAM	-	-	-	-	ALTER
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	-	-	-	-	UPDATE
SYSTEM.CHANNEL.INITQ	-	-	-	-	UPDATE
SYSTEM.CHANNEL.SYNCQ	-	-	-	-	UPDATE
SYSTEM.CLUSTER.COMMAND.QUEUE	-	-	-	-	ALTER
SYSTEM.CLUSTER.REPOSITORY.QUEUE	-	-	-	-	UPDATE
SYSTEM.CLUSTER.TRANSMIT.QUEUE	-	-	-	-	ALTER
SYSTEM.COMMAND.INPUT	UPDATE	-	-	UPDATE	UPDATE
SYSTEM.COMMAND.REPLY.*	-	-	-	-	UPDATE
SYSTEM.COMMAND.REPLY.MODEL	UPDATE	-	-	UPDATE	UPDATE
SYSTEM.CSQOREXX.*	-	-	-	UPDATE	-
SYSTEM.CSQUTIL.*	UPDATE	-	-	-	-
SYSTEM.CSQXCMD.*	-	-	-	-	UPDATE
SYSTEM.HIERARCHY.STATE	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.CONTROL	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.PUBS	-	-	-	-	UPDATE
SYSTEM.INTER.QMGR.FANREQ	-	-	-	-	UPDATE
SYSTEM.PROTECTION.ERROR.QUEUE	-	-	-	-	UPDATE
SYSTEM.PROTECTION.POLICY.QUEUE	-	UPDATE “1” en la página 209	-	-	READ
SYSTEM.QSG.CHANNEL.SYNCQ	-	-	-	-	UPDATE
SYSTEM.QSG.TRANSMIT.QUEUE	-	-	-	-	UPDATE
SYSTEM.REST.REPLY.QUEUE	-	-	UPDATE	-	-
SYSTEM.BLUEMIX.REGISTRATION.QUEUE	-	-	-	-	UPDATE

Notas:

1. El usuario del espacio de direcciones es Advanced Message Security también requiere acceso de lectura (READ) a esta cola.

z/OS *Referencia rápida del acceso a la seguridad de recursos de la API*

Un resumen de las opciones **MQOPEN**, **MQPUT1**, **MQSUB** y **MQCLOSE** y el acceso que necesitan los diferentes tipos de seguridad de recursos.

Tabla 36. Opciones MQOPEN, MQPUT1, MQSUB y MQCLOSE y la autorización de seguridad necesaria. Los comentarios emergentes de tipo **(1)** hacen referencia a las notas que figuran a continuación de esta tabla.

	Nivel de accesoRACF mínimo necesario			
Clase RACF:	**MXTOPIC**	**MQQUEUE o MXQUEUE (1)**	**MQADMIN o MXADMIN**	**MQADMIN o MXADMIN**
Perfil RACF:	**(15 o 16)**	**(2)**	**(3)**	**(4)**
Opción MQOPEN				
MQOO_INQUIRE		READ **(5)**	No se comprueba	No se comprueba
MQOO_BROWSE		READ	No se comprueba	No se comprueba
MQOO_INPUT_*		UPDATE	No se comprueba	No se comprueba
MQOO_SAVE_ALL_CONTEXT **(6)**		UPDATE	No se comprueba	No se comprueba
MQOO_OUTPUT (USAGE=NORMAL) **(7)**		UPDATE	No se comprueba	No se comprueba
MQOO_PASS_IDENTITY_CONTEXT **(8)**		UPDATE	READ	No se comprueba
MQOO_PASS_ALL_CONTEXT **(8)** **(9)**		UPDATE	READ	No se comprueba
MQOO_SET_IDENTITY_CONTEXT **(8)** **(9)**		UPDATE	UPDATE	No se comprueba
MQOO_SET_ALL_CONTEXT **(8)** **(10)**		UPDATE	CONTROL	No se comprueba
MQOO_OUTPUT (USAGE (XMITQ) **(11)**		UPDATE	CONTROL	No se comprueba
MQOO_OUTPUT (objeto de tema)	UPDATE **(16)**			
MQOO_OUTPUT (cola alias a objeto de tema)	UPDATE **(16)**	UPDATE		
MQOO_SET		ALTER	No se comprueba	No se comprueba
MQOO_ALTERNATE_USER_AUTHORITY		**(12)**	**(12)**	UPDATE
Opción MQPUT1				
Transferir a una cola normal **(7)**		UPDATE	No se comprueba	No se comprueba
MQPMO_PASS_IDENTITY_CONTEXT		UPDATE	READ	No se comprueba

Tabla 36. Opciones MQOPEN, MQPUT1, MQSUB y MQCLOSE y la autorización de seguridad necesaria. Los comentarios emergentes de tipo **(1)** hacen referencia a las notas que figuran a continuación de esta tabla. *(continuación)*

	Nivel de accesoRACF mínimo necesario			
Clase RACF:	**MXTOPIC**	**MQQUEUE o MXQUEUE (1)**	**MQADMIN o MXADMIN**	**MQADMIN o MXADMIN**
Perfil RACF:	**(15 o 16)**	**(2)**	**(3)**	**(4)**
MQPMO_PASS_ALL_CONTEXT		UPDATE	READ	No se comprueba
MQPMO_SET_IDENTITY_CONTEXT		UPDATE	UPDATE	No se comprueba
MQPMO_SET_ALL_CONTEXT		UPDATE	CONTROL	No se comprueba
MQOO_OUTPUT Transferir a una cola de transmisión **(11)**		UPDATE	CONTROL	No se comprueba
MQOO_OUTPUT (objeto de tema)	UPDATE **(16)**			
MQOO_OUTPUT (cola alias a objeto de tema)	UPDATE **(16)**	UPDATE		
MQPMO_ALTERNATE_USER_AUTHORITY		**(13)**	**(13)**	UPDATE
Opción `MQCLOSE`				
MQCO_DELETE **(14)**		ALTER	No se comprueba	No se comprueba
MQCO_DELETE_PURGE **(14)**		ALTER	No se comprueba	No se comprueba
MQCO_REMOVE_SUB	ALTER **(15)**			
Opción `MQSUB`				
MQSO_CREATE	ALTER **(15)**	**(17)**	**(18)**	
MQSO_ALTER	ALTER **(15)**	**(17)**	**(18)**	
MQSO_RESUME	READ **(15)**	**(17)**	No se comprueba	
MQSO_ALTERNATE_USER_AUTHORITY				UPDATE
MQSO_SET_IDENTITY_CONTEXT			**(18)**	

Nota:

1. Esta opción no está restringida a colas. Utilice la clase MQNLIST o MXNLIST para listas de nombres y la clase MQPROC o MXPROC para procesos.
2. Utilice el perfil RACF: hlq.resourcename
3. Utilice el perfil RACF: hlq.CONTEXT.queuename
4. Utilice el perfil RACF : hlq.ALTERNATE.USER. `alternateuserid`

   `alternateuserid` es el identificador de usuario que se especifica en el campo *`AlternateUserId`* del descriptor de objeto. Tenga en cuenta que se utilizan hasta 12 caracteres del campo

AlternateUserId para esta comprobación, a diferencia de otras comprobaciones en las que sólo se utilizan los primeros 8 caracteres de un identificador de usuario.

5. No se realiza ninguna comprobación al abrir el gestor de colas para consultas.
6. También debe especificarse MQOO_INPUT_*. Esto es válido para una cola local, modelo o alias.
7. Esta comprobación se realiza para una cola local o modelo que tiene un atributo de cola **Usage** de MQUS_NORMAL, y también para una cola alias o remota (que se define en el gestor de colas conectado.) Si la cola es una cola remota que se abre especificando un *ObjectQMgrName* (no el nombre del gestor de colas conectado) explícitamente, la comprobación se realiza en la cola con el mismo nombre que *ObjectQMgrName* (que debe ser una cola local con un atributo de cola **Usage** de MQUS_TRANSMISSION).
8. También debe especificarse MQOO_OUTPUT.
9. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT.
10. Esta opción también implica MQOO_PASS_IDENTITY_CONTEXT, MQOO_PASS_ALL_CONTEXT y MQOO_SET_IDENTITY_CONTEXT.
11. Esta comprobación se realiza para una cola local o modelo que tiene un atributo de cola **Usage** de MQUS_TRANSMISSION y se está abriendo directamente para salida. No se aplica si se está abriendo una cola remota.
12. También se debe especificar al menos una de las opciones MQOO_INQUIRE, MQOO_BROWSE, MQOO_INPUT_*, MQOO_OUTPUT o MQOO_SET. La comprobación que se lleva a cabo es la misma que para las otras opciones especificadas.
13. La comprobación que se lleva a cabo es la misma que para las otras opciones especificadas.
14. Esto sólo se aplica para colas dinámicas permanentes que se han abierto directamente, es decir, que no se han abierto mediante una cola modelo. No es necesaria ninguna seguridad para suprimir una cola dinámica temporal.
15. Utilice el perfil RACF hlq.SUBSCRIBE.topicname.
16. Utilice el perfil RACF hlq.PUBLISH.topicname.
17. Si en la solicitud MQSUB ha especificado una cola de destino a la que se han de enviar las publicaciones, se lleva a cabo una comprobación de seguridad en esa cola para asegurar que dispone de autorización de transferencia para esa cola.
18. Si en la solicitud MQSUB, con la opción MQSO_CREATE o MQSO_ALTER especificada, desea establecer cualquiera de los campos de contexto de identidad de la estructura MQSD, también tiene que especificar la opción MQSO_SET_IDENTITY_CONTEXT y necesita además la autorización adecuada al perfil de contexto para la cola de destino.

z/OS *Perfiles para la seguridad de temas*

Si la seguridad de temas está activa, debe definir perfiles en las clases correspondientes y dar permiso a los grupos o los ID de usuario necesarios para acceder a dichos perfiles.

El concepto de seguridad de temas dentro de un árbol de temas se describe en Seguridad de publicación/suscripción.

Si la seguridad de temas está activa, debe realizar las siguientes acciones:

- Definir perfiles en las clases **MXTOPIC** o **GMXTOPIC**.
- Permitir a los grupos o los ID de usuario necesarios acceder a estos perfiles, para que puedan emitir peticiones API de IBM MQ que utilicen temas.

Los perfiles para la seguridad de temas adoptan la forma:

```
hlq.SUBSCRIBE.topicname
hlq.PUBLISH.topicname
```

donde

- `hlq` es `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas).
- `topicname` es el nombre del nodo de administración de temas en el árbol de temas, que está asociado al tema al que se está suscribiendo mediante una llamada MQSUB, o en el que se está publicando con una llamada MQOPEN.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el acceso a un único tema en dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el acceso a uno o varios temas con dicho nombre de tema en todos los gestores de colas dentro del grupo de compartición de colas. Este acceso se puede sustituir en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para este tema en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

Suscribir

Para suscribirse a un tema, necesita acceso tanto al tema al que está intentando suscribirse como a la cola de destino de las publicaciones.

Cuando emite una solicitud MQSUB, se llevan a cabo las siguientes comprobaciones de seguridad:

- Si tiene el nivel de acceso adecuado para suscribirse a ese tema y también si la cola de destino (si se especifica) está abierta para salida
- Si tiene el nivel adecuado de acceso a dicha cola de destino.

Tabla 37. Nivel de acceso necesario para la seguridad de temas para suscribirse

Opción MQSUB	**Acceso de RACF necesario para el perfil `hlq.SUBSCRIBE.topicname` en la clase MXTOPIC**
MQSO_CREATE y MQSO_ALTER	ALTER
MQSO_RESUME	READ

Tabla 38. Autorización adicional necesaria para suscribirse utilizando una cola de destino no gestionada

Opción MQSUB	**Se necesita acceso RACF al perfil `hlq.CONTEXT.queuename` en la clase MQADMIN o MXADMIN**
MQSO_CREATE, MQSO_ALTER y MQSO_RESUME	UPDATE
	Se necesita acceso deRACF al perfil `hlq.queuename` en la clase MQQUEUE o MXQUEUE
MQSO_CREATE y MQSO_ALTER	UPDATE
	Se necesita acceso deRACF al perfil `hlq.ALTERNATE.USER.alternateuserid` en la clase MQADMIN o MXADMIN
MQSO_ALTERNATE_USER_AUTHORITY	UPDATE

Consideraciones para las colas gestionadas para suscripciones

Se lleva a cabo una comprobación de seguridad para ver si tiene autorización para suscribirse al tema. Sin embargo, no se lleva a cabo ninguna comprobación de seguridad cuando se crea la cola gestionada, o para determinar si tiene acceso para transferir mensajes a esta cola de destino.

No puede cerrar/suprimir una cola gestionada.

Las colas modelo utilizadas son: SYSTEM.DURABLE.MODEL.QUEUE y SYSTEM.NDURABLE.MODEL.QUEUE.

Las colas gestionadas que se crean a partir de estas colas modelo adoptan la forma SYSTEM.MANAGED.DURABLE.A346EF00367849A0 y SYSTEM.MANAGED.NDURABLE.A346EF0036785EA0, donde el último calificador es imprevisible.

No otorgue a ningún usuario acceso a estas colas. Las colas se pueden proteger utilizando perfiles genéricos de la forma SYSTEM.MANAGED.DURABLE.* y SYSTEM.MANAGED.NDURABLE.* sin ninguna autorización otorgada.

Se pueden recuperar mensajes de estas colas utilizando el manejador devuelto en la solicitud MQSUB.

Si emite explícitamente una llamada MQCLOSE para una suscripción con la opción MQCO_REMOVE_SUB especificada, y no ha creado la suscripción que está cerrando bajo este manejador, se realiza una comprobación de seguridad en el momento del cierre para asegurar que dispone de la autorización correcta para realizar la operación.

Tabla 39. Nivel de acceso necesario a los perfiles de seguridad de temas para la operación de cierre de una suscripción

Opción MQCLOSE	**Acceso de RACF necesario para el perfil `hlq.SUBSCRIBE.topicname` en la clase MXTOPIC**
MQCO_REMOVE_SUB	ALTER

Publicar

Para publicar en un tema, necesita tener acceso al tema y, si está utilizando colas alias, también a la cola alias.

Tabla 40. Nivel de acceso necesario para que se publique la seguridad de tema

Opción MQOPEN o MQPUT1	**Acceso de RACF necesario para el perfil `hlq.PUBLISH.topicname` en la clase MXTOPIC**
MQOO_OUTPUT o MQPUT1	UPDATE

Tabla 41. Nivel de acceso necesario para abrir una cola alias que se resuelve en un tema

Opción MQOPEN o MQPUT1	**Acceso RACF necesario al perfil `hlq.queuename` en la clase MQQUEUE o MXQUEUE para la cola alias**
MQOO_OUTPUT o MQPUT1	UPDATE

Para obtener más información sobre cómo funciona la seguridad de temas cuando una cola alias que se resuelve en un nombre de tema se abre para publicación, consulte “Consideraciones para las colas alias que se resuelven en temas para una operación de publicación” en la página 214.

Cuando considere colas alias utilizadas para colas de destino para restricciones PUT o GET, consulte “Consideraciones para las colas alias” en la página 203.

Si se cambia el nivel de acceso RACF que una aplicación tiene a un perfil de seguridad de temas, los cambios sólo tienen efecto para los nuevos manejadores de objeto obtenidos (es decir, una nueva MQSUB o MQOPEN) para dicho tema. Los manejadores que ya existían en el momento del cambio mantienen su acceso existente al tema. Asimismo, los suscriptores existentes mantienen su acceso a todas las suscripciones que ya han hecho.

Consideraciones para las colas alias que se resuelven en temas para una operación de publicación

Cuando se emite una llamada MQOPEN o MQPUT1 para una cola alias que se resuelve en un tema, IBM MQ realiza dos comprobaciones de recursos:

- La primera en el nombre de cola alias especificado en el descriptor de objeto (MQOD) de la llamada MQOPEN o MQPUT1.
- La segunda en el tema en que se resuelve la cola alias

Debe tener en cuenta que este comportamiento es diferente del que se obtiene cuando las colas alias se resuelven en otras colas. Necesita el acceso correcto a ambos perfiles para que la acción de publicación pueda continuar.

Seguridad de temas del sistema

El espacio de direcciones del iniciador de canal accede a los siguientes temas del sistema.

Los ID de usuario bajo los que se ejecuta esto deben tener acceso RACF a estas colas, tal como se muestra en la Tabla 42 en la página 214.

Tabla 42. Acceso necesario a los temas SYSTEM

Tema SYSTEM	Perfil	Iniciador de canal para gestión de colas distribuidas
SYSTEM.BROKER.ADMIN.STREAM	hlq.PUBLISH.topicname	UPDATE
SYSTEM.BROKER.ADMIN.STREAM	hlq.SUBSCRIBE.topicname	ALTER

z/OS *Perfiles para procesos*

Si la seguridad de procesos está activa, debe definir perfiles en las clases adecuadas y permitir a los grupos o los ID de usuario necesarios acceder a dichos perfiles.

Si la seguridad de procesos está activa, debe:

- Definir perfiles en las clases **MQPROC** o **GMQPROC** si se utilizan perfiles en mayúsculas.
- Definir perfiles en las clases **MXPROC** o **GMXPROC** si se utilizan perfiles que combinan mayúsculas y minúsculas.
- Permitir a los grupos o los ID de usuario necesarios acceder a estos perfiles, para que puedan emitir peticiones API de IBM MQ que utilicen procesos.

Los perfiles para procesos adoptan la forma:

```
hlq.processname
```

donde `hlq` puede ser `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas), y `processname` es el nombre del proceso que se abre.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el acceso a una única definición de proceso de dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el acceso a una o varias definiciones de proceso con dicho nombre en todos los gestores de colas dentro del grupo de compartición de colas. en todos los gestores de colas dentro del grupo de compartición de colas. Este acceso se pueden sustituir en un gestor de colas individual mediante la definición de un perfil de nivel de gestor de colas para esa definición de proceso en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil

que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

La tabla siguiente muestra el acceso necesario para abrir un proceso.

Tabla 43. Niveles de acceso para seguridad de procesos

Opción MQOPEN	**Nivel de acceso de RACF necesario para hlq.processname**
MQOO_INQUIRE	READ

Por ejemplo, en el gestor de colas MQS9, el grupo RACF INQVPRC debe poder consultar (`MQINQ`) en todos los procesos que empiezan con la letra V. Las definiciones de RACF para esto serían:

```
RDEFINE MQPROC MQS9.V* UACC(NONE)
PERMIT MQS9.V* CLASS(MQPROC) ID(INQVPRC) ACCESS(READ)
```

La seguridad de usuario alternativo también puede estar activa, dependiendo de las opciones de apertura que se especifiquen al abrir un objeto de definición de proceso.

z/OS *Perfiles para listas de nombres*

Si la seguridad de listas de nombres está activa, defina perfiles en las clases adecuadas y permita a los grupos o los ID de usuario necesarios acceder a dichos perfiles.

Si la seguridad de listas de nombres está activa, debe:

- Definir perfiles en las clases **MQNLIST** o **GMQNLIST** si se utilizan perfiles en mayúsculas.
- Definir perfiles en las clases **MXNLIST** o **GMXNLIST** si se utilizan perfiles que combinan mayúsculas y minúsculas.
- Permitir a los grupos o los ID de usuario necesarios acceder a estos perfiles.

Los perfiles para listas de nombres adoptan la forma:

```
hlq.namelistname
```

donde `hlq` puede ser `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas), y `namelistname` es el nombre de la lista de nombres que se abre.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el acceso a una única lista de nombres de dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el acceso a una o varias definiciones de proceso con dicho nombre en todos los gestores de colas dentro del grupo de compartición de colas. en todos los gestores de colas dentro del grupo de compartición de colas. Este acceso se puede modificar en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para esa lista de nombres en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

La tabla siguiente muestra el acceso necesario para abrir una lista de nombres.

Tabla 44. Niveles de acceso para seguridad de listas de nombres

Opción MQOPEN	**Nivel de acceso RACF necesario para hlq.namelistname**
MQOO_INQUIRE	READ

Por ejemplo, en el gestor de colas (o grupo de compartición de colas) PQM3, el grupo RACF DEPT571 debe poder consultar (MQINQ) en estas listas de nombres:

- Todas las listas de nombres que empiezan por "DEPT571".
- PRINTER/DESTINATIONS/DEPT571
- AGENCY/REQUEST/QUEUES
- WAREHOUSE.BROADCAST

Las definiciones RACF para hacer esto son:

```
RDEFINE MQNLIST PQM3.DEPT571.** UACC(NONE)
PERMIT PQM3.DEPT571.** CLASS(MQNLIST) ID(DEPT571) ACCESS(READ)

RDEFINE GMQNLIST NLISTS.FOR.DEPT571 UACC(NONE)
        ADDMEM(PQM3.PRINTER/DESTINATIONS/DEPT571,
               PQM3.AGENCY/REQUEST/QUEUES,
               PQM3.WAREHOUSE.BROADCAST)
PERMIT NLISTS.FOR.DEPT571 CLASS(GMQNLIST) ID(DEPT571) ACCESS(READ)
```

La seguridad de usuario alternativo puede estar activa, dependiendo de las opciones que se especifiquen al abrir un objeto de lista de nombres.

Seguridad de listas de nombres del sistema

Muchas de las listas de nombres del sistema son accedidas por los componentes auxiliares de IBM MQ:

- El programa de utilidad CSQUTIL
- Los paneles de operaciones y los paneles de control
- El espacio de direcciones del iniciador de canal (incluido el daemon de publicación/suscripción en cola)

Los ID de usuario con los que se ejecutan deben tener acceso RACF a estas listas de nombres, tal como se muestra en la Tabla 45 en la página 216.

Tabla 45. Acceso necesario a las listas de nombres SYSTEM por parte de IBM MQ

Lista de nombres SYSTEM	CSQUTIL	Paneles de operaciones y paneles de control	Iniciador de canal para gestión de colas distribuidas
SYSTEM.QPUBSUB.QUEUE.NAMELIST	-	-	READ
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	-	-	READ

z/OS *Perfiles para la seguridad de usuario alternativo*

Si la seguridad de usuario alternativo está activa, debe definir perfiles en las clases adecuadas y permitir a los grupos o los ID de usuario necesarios acceder a dichos perfiles.

Para obtener más información sobre *AlternateUserId*, consulte AlternateUserID (MQCHAR12).

Si la seguridad de usuario alternativo está activa, debe:

- Definir perfiles en las clases MQADMIN o GMQADMIN si está utilizando perfiles en mayúsculas.
- Definir perfiles en las clases MXADMIN o GMXADMIN si está utilizando perfiles que combinan mayúsculas y minúsculas.

Permitir a los grupos o los ID de usuario necesarios acceder a estos perfiles, para que puedan utilizar las opciones ALTERNATE_USER_AUTHORITY cuando se abra el objeto.

Los perfiles para seguridad de usuario alternativo se pueden especificar a nivel de subsistema o a nivel de grupo de compartición de colas, y adoptan la forma siguiente:

```
hlq.ALTERNATE.USER.alternateuserid
```

Donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas), y `alternateuserid` es el valor del campo *AlternateUserId* en el descriptor de objeto.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el uso de un ID de usuario alternativo en dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el uso de un ID de usuario alternativo en todos los gestores de colas del grupo de compartición de colas. Un usuario que tenga el acceso correcto puede utilizar este ID de usuario alternativo en cualquier gestor de colas del grupo de compartición de colas. Este acceso se puede modificar en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para dicho ID de usuario alternativo en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

La tabla siguiente muestra el acceso necesario al especificar una opción de usuario alternativo.

Tabla 46. Niveles de acceso para la seguridad de usuario alternativo

Opción MQOPEN, MQSUB o MQPUT1	**Nivel de acceso RACF necesario**
MQOO_ALTERNATE_USER_AUTHORITY MQSO_ALTERNATE_USER_AUTHORITY MQPMO_ALTERNATE_USER_AUTHORITY	UPDATE

Además de las comprobaciones de seguridad de usuario alternativo, también se pueden realizar otras comprobaciones de seguridad para la seguridad de colas, de procesos, de listas de nombres y de contexto. El ID de usuario alternativo, si se proporciona, sólo se utiliza para comprobaciones de seguridad en recursos de colas, definiciones de proceso y listas de nombres. Para las comprobaciones de seguridad de usuario alternativo y de contexto, se utiliza el ID de usuario que solicita la comprobación. Para obtener información detallada sobre cómo se manejan los ID de usuario, consulte “ID de usuario para la comprobación de seguridad en z/OS” en la página 240. Para ver una tabla de resumen que muestra las opciones de apertura y las comprobaciones de seguridad necesarias cuando la seguridad de colas, de contexto y de usuario alternativo están todas activas, consulte Tabla 36 en la página 209.

Un perfil de usuario alternativo otorga al ID de usuario solicitante acceso a los recursos asociados al ID de usuario especificado en el ID de usuario alternativo. Por ejemplo, el servidor de nóminas que se ejecuta bajo el ID de usuario PAYSERV en el gestor de colas QMPY procesa las peticiones de los ID de usuario de personal, que empiezan todos por PS. Para hacer que el trabajo realizado por el servidor de nóminas se lleve a cabo bajo el ID de usuario del usuario solicitante, se utiliza autorización de usuario alternativo. El servidor de nóminas sabe qué ID de usuario ha de especificar como el ID de usuario alternativo porque los programas solicitantes generan mensajes utilizando la opción de transferencia de mensaje MQPMO_DEFAULT_CONTEXT. Consulte “ID de usuario para la comprobación de seguridad en z/OS” en la página 240 para obtener más detalles acerca de dónde se obtienen los ID de usuario alternativo.

Las siguientes definiciones RACF de ejemplo permiten al programa servidor especificar identificadores (ID) de usuario alternativo que empiecen por los caracteres PS:

```
RDEFINE MQADMIN QMPY.ALTERNATE.USER.PS* UACC(NONE)
PERMIT QMPY.ALTERNATE.USER.PS* CLASS(MQADMIN) ID(PAYSERV) ACCESS(UPDATE)
```

Nota:

1. Los campos *AlternateUserId* del descriptor de objetos y el descriptor de suscripciones tienen una longitud de 12 bytes. Los 12 bytes se utilizan en las comprobaciones de perfil, pero IBM MQ. Si

este truncamiento del ID de usuario no es conveniente, los programas de aplicación que realizan la solicitud deben convertir cualquier ID de usuario alternativo que tenga más de 8 bytes en algo más adecuado.

2. Si especifica MQOO_ALTERNATE_USER_AUTHORITY, MQSO_ALTERNATE_USER_AUTHORITY o MQPMO_ALTERNATE_USER_AUTHORITY y no especifica un campo *AlternateUserId* en el descriptor de objeto, se utiliza un ID de usuario de blancos. A efectos de la comprobación de seguridad de usuario alternativo, el ID de usuario utilizado para el calificador *AlternateUserId* es -BLANK-. Por ejemplo, `RDEF MQADMIN hlq.ALTERNATE.USER.-BLANK-`.

   Si el usuario tiene permiso para acceder a este perfil, todas las comprobaciones posteriores se realizan con un ID de usuario de espacios en blanco. Para obtener información detallada sobre los ID de usuario de espacios en blanco, consulte "Identificadores de usuario en blanco y niveles de UACC" en la página 249.

La administración de los ID de usuario alternativo es más fácil si tiene un convenio de denominación para los ID de usuario que le permita utilizar perfiles de usuario alternativo genéricos. De no ser así, puede utilizar la característica RACVARS de RACF RACVARS. Para obtener detalles sobre cómo utilizar RACVARS, consulte la publicación *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

Cuando un mensaje se transfiere a una cola que se ha abierto con autorización de usuario alternativo, y el contexto del mensaje ha sido generado por el gestor de colas, el campo MQMD_USER_IDENTIFIER se establece en el ID de usuario alternativo.

z/OS *Perfiles para la seguridad de contexto*

IBM MQ utiliza perfiles para controlar el acceso a la información de contexto específica de un mensaje determinado. El contexto está contenido en el descriptor de mensaje (MQMD).

Utilización de perfiles para la seguridad de contexto

Si la seguridad de contexto está activa, debe:

- Definir un perfil en la clase **MQADMIN** si se utilizan perfiles en mayúsculas.
- Definir un perfil en la clase **MXADMIN** si se utilizan perfiles que combinan mayúsculas y minúsculas.

El perfil se denomina `hlq.CONTEXT.queuename` o `hlq.CONTEXT.topicname`, donde:

hlq
Puede ser `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas).

queuename
Puede ser el nombre completo de la cola para la que desea definir el perfil de contexto, o un perfil genérico.

TOPICNAME
Puede ser el nombre completo del tema para el que desea definir el perfil de contexto o un perfil genérico.

Un perfil con el prefijo del nombre del gestor de colas, y con ** especificado como nombre de cola o tema, permite el control de la seguridad de contexto en todas las colas y temas que pertenecen a ese gestor de colas. Esto se puede alterar temporalmente en una cola o tema individual definiendo un perfil específico para el contexto en esa cola o tema.

Un perfil con el prefijo del nombre del grupo de compartición de colas, y con ** especificado como nombre de cola o tema, permite el control del contexto en todas las colas y temas que pertenecen a los gestores de colas dentro del grupo de compartición de colas. Esto se puede sustituir en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para el contexto en dicho gestor de colas, especificando un perfil con el nombre del gestor de colas como prefijo. También se puede alterar temporalmente en una cola o tema individual especificando un perfil con el sufijo del nombre de cola o tema.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

Debe otorgar a los grupos o los ID de usuario necesarios acceso a este perfil. La tabla siguiente muestra el nivel de acceso necesario, que depende de la especificación de las opciones de contexto cuando se abre la cola.

Tabla 47. Niveles de acceso para seguridad de contexto

Opción MQOPEN o MQPUT1	**RACF nivel de acceso necesario para hlq.CONTEXT.queuename o hlq.CONTEXT.topicname**
MQPMO_NO_CONTEXT	No hay comprobación de seguridad de contexto
MQPMO_DEFAULT_CONTEXT	No hay comprobación de seguridad de contexto
MQOO_SAVE_ALL_CONTEXT	No hay comprobación de seguridad de contexto
MQOO_PASS_IDENTITY_CONTEXT MQPMO_PASS_IDENTITY_CONTEXT	READ
MQOO_PASS_ALL_CONTEXT MQPMO_PASS_ALL_CONTEXT	READ
MQOO_SET_IDENTITY_CONTEXT MQPMO_SET_IDENTITY_CONTEXT	UPDATE
MQOO_SET_ALL_CONTEXT MQPMO_SET_ALL_CONTEXT	CONTROL
MQOO_OUTPUT o MQPUT1(USAGE(XMITQ))	CONTROL
Opción MQSUB	
MQSO_SET_IDENTITY_CONTEXT **(Nota 2)**	UPDATE

Nota:

1. Los ID de usuario que se utilizan para la gestión de colas distribuidas necesitan acceso CONTROL a `hlq.CONTEXT.queuename` para transferir mensaje a la cola de destino. Consulte "Identificadores de usuario utilizados por el iniciador de canal" en la página 243 para obtener información sobre los ID de usuario utilizados.
2. Si en la solicitud MQSUB, con la opción MQSO_CREATE o MQSO_ALTER especificada, desea establecer cualquiera de los campos de contexto de identidad de la estructura MQSD, tiene que especificar la opción MQSO_SET_IDENTITY_CONTEXT. También necesita la autorización adecuada al perfil de contexto para la cola de destino.

Si transfiere mandatos a la cola de entrada de mandatos del sistema, utilice la opción de transferencia de mensaje de contexto predeterminado para asociar el ID de usuario correcto al mandato.

Por ejemplo, el programa de utilidad CSQUTIL suministrado por IBM MQ se puede utilizar para descargar y recargar mensajes en colas. Cuando los mensajes descargados se restauran en una cola, el programa de utilidad CSQUTIL utiliza la opción MQOO_SET_ALL_CONTEXT para devolver los mensajes a su estado original. Además de la seguridad de colas que requiere esta opción de apertura, también se requiere autorización de contexto. Por ejemplo, si el grupo BACKGRP en el gestor de colas MQS1 requiere esta autorización, esto se define mediante:

```
RDEFINE MQADMIN MQS1.CONTEXT.** UACC(NONE)
PERMIT MQS1.CONTEXT.** CLASS(MQADMIN) ID(BACKGRP) ACCESS(CONTROL)
```

En función de las opciones especificadas, y de los tipos de seguridad realizados, también pueden tener lugar otros tipos de comprobaciones de seguridad cuando se abre la cola. Estos incluyen la seguridad de colas (consulte "Perfiles para la seguridad de colas" en la página 201) y la seguridad de usuario alternativo (consulte "Perfiles para la seguridad de usuario alternativo" en la página 216). Para ver una tabla de resumen que muestra las opciones de apertura y las comprobaciones de seguridad necesarias cuando la seguridad de colas, de contexto y de usuario alternativo están todas activas, consulte Tabla 36 en la página 209.

Seguridad de contexto de colas del sistema

Muchas de las colas del sistema pueden ser accedidas por las partes auxiliares de IBM MQ, por ejemplo, el estado de direcciones del iniciador del canal V 9.1.0, y el servidor mqweb utilizado por IBM MQ Console y REST API.

A los ID de usuario bajo los cuales se ejecutan estos se les debe proporcionar acceso RACF a estas colas, tal como se indica en Tabla 48 en la página 220.

Tabla 48. Acceso necesario a las colas SYSTEM para operaciones de contexto

Cola SYSTEM	**Iniciador de canal para gestión de colas distribuidas**	**Servidor mqweb**
SYSTEM.ADMIN.COMMAND.QUEUE	-	CONTROL
SYSTEM.BROKER.CONTROL.QUEUE	CONTROL	-
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	CONTROL	-
SYSTEM.CHANNEL.SYNCQ	CONTROL	-
SYSTEM.CLUSTER.COMMAND.QUEUE	CONTROL	-
SYSTEM.CLUSTER.TRANSMIT.QUEUE	CONTROL	-

z/OS *Perfiles para la seguridad de mandatos*

Para permitir la comprobación de seguridad de los mandatos, añada perfiles a la clase MQCMDS. Los nombres de perfil están basados en los mandatos MQSC, pero controlan tanto mandatos MQSC como PCF. Los perfiles se pueden aplicar a un gestor de colas o a un grupo de compartición de colas.

Si desea comprobación de seguridad para los mandatos (por lo que no ha definido el perfil de conmutador de seguridad de mandatos hlq.NO.CMD.CHECKS), debe añadir perfiles a la clase MQCMDS.

Los mismos perfiles de seguridad controlan tanto mandatos MQSC como PCF. Los nombres de los perfiles RACF para la comprobación de seguridad de mandatos se basan en los propios mandatos MQSC. Estos perfiles adoptan la forma:

```
hlq.verb.pkw
```

Donde `hlq` puede ser `qmgr-name` (nombre del gestor de colas) o `qsg-name` (nombre del grupo de compartición de colas), `verb` es la parte del nombre de mandato correspondiente al verbo, por ejemplo ALTER, y `pkw` es el tipo de objeto, por ejemplo QLOCAL para una cola local.

Por tanto, el nombre de perfil para el mandato ALTER QLOCAL en el subsistema CSQ1 es:

```
CSQ1.ALTER.QLOCAL
```

Puede utilizar perfiles genéricos para proteger conjuntos de mandatos, de manera que necesite mantener menos perfiles y, por tanto, menos listas de acceso. Considere la posibilidad de crear un perfil genérico que se aplique a todos los mandatos que no estén protegidos por un perfil más especifico. Defina este perfil con UACC(NONE) y otorgue acceso ALTER sólo a los grupos RACF que contengan administradores. Luego puede crear un perfil genérico aplicable a todos los mandatos DISPLAY y otorgar acceso general al mismo. Como alternativa intermedia, puede identificar grupos de usuarios que necesiten acceso a ciertos conjuntos de mandatos, en cuyo caso podría crear perfiles para esos conjuntos y otorgar acceso a grupos RACF que representen esas clases de usuario. Evite otorgar a los usuarios acceso a mandatos que no necesiten: aplique el principio de privilegio mínimo, para que los usuarios sólo tengan acceso a los mandatos que necesiten para sus trabajos.

Un perfil que lleve como prefijo el nombre del gestor de colas controla el uso del mandato en dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el uso del mandato en todos los gestores de colas del grupo de compartición de colas. Este acceso se puede alterar temporalmente en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para este mandato en dicho gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

Mediante la configuración de perfiles de mandato a nivel de gestor de colas, se puede impedir a un usuario emitir mandatos en un gestor de colas determinado. También puede definir un perfil para un grupo de compartición de colas para cada verbo de mandato, y todas las comprobaciones de seguridad tendrán lugar en dicho perfil, en lugar de en gestores de colas individuales.

Si tanto la seguridad de subsistema como la seguridad de grupo de compartición de colas están activas y no se encuentra un perfil local, se realiza una comprobación de seguridad de mandatos para ver si el usuario tiene acceso a un perfil de grupo de compartición de colas.

Si utiliza el atributo CMDSCOPE para encaminar un mandato hacia otros gestores de colas de un grupo de compartición de colas, se realiza una comprobación de seguridad en cada gestor de colas en el que se ejecuta el mandato, pero no necesariamente en el gestor de colas en el que se emite el mandato.

La Tabla 49 en la página 221 muestra, para cada mandato MQSC de IBM MQ , los perfiles necesarios para que se lleve a cabo la comprobación de seguridad de mandatos y el nivel de acceso correspondiente para cada perfil de la clase MQCMDS.

Tabla 50 en la página 226 muestra, para cada mandato PCF de IBM MQ, los perfiles necesarios para llevar a cabo la comprobación de seguridad de mandatos, y el nivel de acceso correspondiente para cada perfil de la clase MQCMDS.

Tabla 49. Mandatos MQSC, perfiles y sus niveles de acceso

Mandato	**Perfil de mandato para MQCMDS**	**Nivel de acceso para MQCMDS**	**Perfil de recurso de mandato para MQADMIN o MXADMIN**	**Nivel de acceso para MQADMIN o MXADMIN**
ALTER AUTHINFO	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
ALTER BUFFPOOL	hlq.ALTER.BUFFPOOL	ALTER	No se comprueba	-
ALTER CFSTRUCT	hlq.ALTER.CFSTRUCT	ALTER	No se comprueba	-
ALTER CHANNEL	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
ALTER NAMELIST	hlq.ALTER.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER

Tabla 49. Mandatos MQSC, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
ALTER PROCESS	hlq.ALTER.PROCESS	ALTER	hlq.PROCESS.process	ALTER
ALTER PSID	hlq.ALTER.PSID	ALTER	No se comprueba	-
ALTER QALIAS	hlq.ALTER.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
ALTER QLOCAL	hlq.ALTER.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
ALTER QMGR	hlq.ALTER.QMGR	ALTER	No se comprueba	-
ALTER QMODEL	hlq.ALTER.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
ALTER QREMOTE	hlq.ALTER.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER
ALTER SECURITY	hlq.ALTER.SECURITY	ALTER	No se comprueba	-
ALTER SMDS	hlq.ALTER.SMDS	ALTER	No se comprueba	-
ALTER STGCLASS	hlq.ALTER.STGCLASS	ALTER	No se comprueba	-
ALTER SUB	hlq.ALTER.SUB	ALTER	No se comprueba	-
ALTER TOPIC	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
ALTER TRACE	hlq.ALTER.TRACE	ALTER	No se comprueba	-
ARCHIVE LOG	hlq.ARCHIVE.LOG	CONTROL	No se comprueba	-
BACKUP CFSTRUCT	hlq.BACKUP.CFSTRUCT	CONTROL	No se comprueba	-
CLEAR QLOCAL	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
CLEAR TOPICSTR “3” en la página 226	hlq.CLEAR.TOPICSTR	ALTER	hlq.TOPIC.topic	ALTER
DEFINE AUTHINFO	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
DEFINE BUFFPOOL	hlq.DEFINE.BUFFPOOL	ALTER	No se comprueba	-
DEFINE CFSTRUCT	hlq.DEFINE.CFSTRUCT	ALTER	No se comprueba	-
DEFINE CHANNEL	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
DEFINE LOG	hlq.DEFINE.LOG	ALTER	No se comprueba	-
DEFINE MAXSMSGS	hlq.DEFINE.MAXSMSGS	ALTER	No se comprueba	-
DEFINE NAMELIST	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
DEFINE PROCESS	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
DEFINE PSID	hlq.DEFINE.PSID	ALTER	No se comprueba	-
DEFINE QALIAS	hlq.DEFINE.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
DEFINE QLOCAL	hlq.DEFINE.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
DEFINE QMODEL	hlq.DEFINE.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
DEFINE QREMOTE	hlq.DEFINE.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER
DEFINE STGCLASS	hlq.DEFINE.STGCLASS	ALTER	No se comprueba	-

Tabla 49. Mandatos MQSC, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
DEFINE SUB	hlq.DEFINE.SUB	ALTER	No se comprueba	-
DEFINE TOPIC	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
DELETE AUTHINFO	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
DELETE BUFFPOOL	hlq.DELETE.BUFFPOOL	ALTER	No se comprueba	-
DELETE CFSTRUCT	hlq.DELETE.CFSTRUCT	ALTER	No se comprueba	-
DELETE CHANNEL	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
DELETE NAMELIST	hlq.DELETE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
DELETE PROCESS	hlq.DELETE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
DELETE PSID	hlq.DELETE.PSID	ALTER	No se comprueba	-
DELETE QALIAS	hlq.DELETE.QALIAS	ALTER	hlq.QUEUE.queue	ALTER
DELETE QLOCAL	hlq.DELETE.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
DELETE QMODEL	hlq.DELETE.QMODEL	ALTER	hlq.QUEUE.queue	ALTER
DELETE QREMOTE	hlq.DELETE.QREMOTE	ALTER	hlq.QUEUE.queue	ALTER
DELETE STGCLASS	hlq.DELETE.STGCLASS	ALTER	No se comprueba	-
DELETE SUB	hlq.DELETE.SUB	ALTER	No se comprueba	-
DELETE TOPIC	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
DISPLAY ARCHIVE “1” en la página 226	hlq.DISPLAY.ARCHIVE	READ	No se comprueba	-
DISPLAY AUTHINFO	hlq.DISPLAY.AUTHINFO	READ	No se comprueba	-
DISPLAY CFSTATUS	hlq.DISPLAY.CFSTATUS	READ	No se comprueba	-
DISPLAY CFSTRUCT	hlq.DISPLAY.CFSTRUCT	READ	No se comprueba	-
DISPLAY CHANNEL	hlq.DISPLAY.CHANNEL	READ	No se comprueba	-
DISPLAY CHINIT	hlq.DISPLAY.CHINIT	READ	No se comprueba	-
DISPLAY CHLAUTH	hlq.DISPLAY.CHLAUTH	READ	No se comprueba	-
DISPLAY CHSTATUS	hlq.DISPLAY.CHSTATUS	READ	No se comprueba	-
DISPLAY CLUSQMGR	hlq.DISPLAY.CLUSQMGR	READ	No se comprueba	-
DISPLAY CMDSERV	hlq.DISPLAY.CMDSERV	READ	No se comprueba	-
DISPLAY CONN “1” en la página 226	hlq.DISPLAY.CONN	READ	No se comprueba	-
DISPLAY GROUP	hlq.DISPLAY.GROUP	READ	No se comprueba	-
DISPLAY LOG “1” en la página 226	hlq.DISPLAY.LOG	READ	No se comprueba	-
DISPLAY MAXSMSGS	hlq.DISPLAY.MAXSMSGS	READ	No se comprueba	-

Tabla 49. Mandatos MQSC, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
DISPLAY NAMELIST	hlq.DISPLAY.NAMELIST	READ	No se comprueba	-
DISPLAY PROCESS	hlq.DISPLAY.PROCESS	READ	No se comprueba	-
DISPLAY PUBSUB	hlq.DISPLAY.PUBSUB	READ	No se comprueba	-
DISPLAY QALIAS	hlq.DISPLAY.QALIAS	READ	No se comprueba	-
DISPLAY QCLUSTER	hlq.DISPLAY.QCLUSTER	READ	No se comprueba	-
DISPLAY QLOCAL	hlq.DISPLAY.QLOCAL	READ	No se comprueba	-
DISPLAY QMGR	hlq.DISPLAY.QMGR	READ	No se comprueba	-
DISPLAY QMODEL	hlq.DISPLAY.QMODEL	READ	No se comprueba	-
DISPLAY QREMOTE	hlq.DISPLAY.QREMOTE	READ	No se comprueba	-
DISPLAY QSTATUS	hlq.DISPLAY.QSTATUS	READ	No se comprueba	-
DISPLAY QUEUE	hlq.DISPLAY.QUEUE	READ	No se comprueba	-
DISPLAY SBSTATUS	hlq.DISPLAY.SBSTATUS	READ	No se comprueba	-
DISPLAY SMDS	hlq.DISPLAY.SMDS	READ	No se comprueba	-
DISPLAY SMDSCONN	hlq.DISPLAY.SMDSCONN	READ	No se comprueba	-
DISPLAY SUB	hlq.DISPLAY.SUB	READ	No se comprueba	-
DISPLAY SECURITY	hlq.DISPLAY.SECURITY	READ	No se comprueba	-
DISPLAY STGCLASS	hlq.DISPLAY.STGCLASS	READ	No se comprueba	-
DISPLAY SYSTEM “1” en la página 226	hlq.DISPLAY.SYSTEM	READ	No se comprueba	-
DISPLAY THREAD	hlq.DISPLAY.THREAD	READ	No se comprueba	-
DISPLAY TPSTATUS	hlq.DISPLAY.TPSTATUS	READ	No se comprueba	-
DISPLAY TOPIC	hlq.DISPLAY.TOPIC	READ	No se comprueba	-
DISPLAY TPSTATUS	hlq.DISPLAY.TPSTATUS	READ	No se comprueba	-
DISPLAY TRACE	hlq.DISPLAY.TRACE	READ	No se comprueba	-
DISPLAY USAGE “1” en la página 226	hlq.DISPLAY.USAGE	READ	No se comprueba	-
MOVE QLOCAL	hlq.MOVE.QLOCAL	ALTER	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	ALTER
PING CHANNEL	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
RECOVER BSDS	hlq.RECOVER.BSDS	CONTROL	No se comprueba	-
RECOVER CFSTRUCT	hlq.RECOVER.CFSTRUCT	CONTROL	No se comprueba	-
REFRESH CLUSTER	hlq.REFRESH.CLUSTER	ALTER	No se comprueba	-
REFRESH QMGR	hlq.REFRESH.QMGR	ALTER	No se comprueba	-

Tabla 49. Mandatos MQSC, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
REFRESH SECURITY	hlq.REFRESH.SECURITY	ALTER	No se comprueba	-
RESET CFSTRUCT	hlq.RESET.CFSTRUCT	CONTROL	No se comprueba	-
RESET CHANNEL	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
RESET CLUSTER	hlq.RESET.CLUSTER	CONTROL	No se comprueba	-
RESET QMGR	hlq.RESET.QMGR	CONTROL	No se comprueba	-
RESET QSTATS	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
RESET SMDS	hlq.RESET.SMDS	CONTROL	No se comprueba	-
RESET TPIPE	hlq.RESET.TPIPE	CONTROL	No se comprueba	-
RESOLVE CHANNEL	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
RESOLVE INDOUBT	hlq.RESOLVE.INDOUBT	CONTROL	No se comprueba	-
RESUME QMGR	hlq.RESUME.QMGR	CONTROL	No se comprueba	-
RVERIFY SECURITY	hlq.RVERIFY.SECURITY	ALTER	No se comprueba	-
SET ARCHIVE	hlq.SET.ARCHIVE	CONTROL	No se comprueba	-
SET CHLAUTH	hlq.SET.CHLAUTH	CONTROL	No se comprueba	-
SET LOG	hlq.SET.LOG	CONTROL	No se comprueba	-
SET SYSTEM	hlq.SET.SYSTEM	CONTROL	No se comprueba	-
START CHANNEL	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
START CHINIT “4” en la página 226	hlq.START.CHINIT	CONTROL	No se comprueba	-
START CMDSERV	hlq.START.CMDSERV	CONTROL	No se comprueba	-
START LISTENER	hlq.START.LISTENER	CONTROL	No se comprueba	-
START QMGR	Ninguno “2” en la página 226	-	-	-
START SMDSCONN	hlq.START.SMDSCONN	CONTROL	No se comprueba	-
START TRACE	hlq.START.TRACE	CONTROL	No se comprueba	-
STOP CHANNEL	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
STOP CHINIT	hlq.STOP.CHINIT	CONTROL	No se comprueba	-
STOP CMDSERV	hlq.STOP.CMDSERV	CONTROL	No se comprueba	-
STOP LISTENER	hlq.STOP.LISTENER	CONTROL	No se comprueba	-
STOP QMGR	hlq.STOP.QMGR	CONTROL	No se comprueba	-
STOP SMDSCONN	hlq.STOP.SMDSCONN	CONTROL	No se comprueba	-
STOP TRACE	hlq.STOP.TRACE	CONTROL	No se comprueba	-
SUSPEND QMGR	hlq.SUSPEND.QMGR	CONTROL	No se comprueba	-

Notas:

1. Estos mandatos puede emitirlos internamente el gestor de colas; no se comprueba ninguna autorización en estos casos.
2. IBM MQ no comprueba la autorización del usuario que emite el mandato START QMGR. Sin embargo, puede utilizar RACF, o recursos alternativos de seguridad para controlar el acceso al mandato START xxxxMSTR que se emite como resultado del mandato START QMGR. Esto se lleva a cabo controlando el acceso al perfil MVS.START.STC.xxxxMSTR en la clase de mandatos de operador RACF (OPERCMDS). Para conocer detalles sobre este procedimiento, consulte la publicación *z/OS SecureWay Security Server RACF Security Administrator's Guide*. Si utiliza esta técnica, y un usuario no autorizado intenta iniciar el gestor de colas, éste termina con un código de razón 00F30216.
3. El recurso **hlq.TOPIC.topic** hace referencia al objeto de tema derivado de TOPICSTR. Para obtener más detalles, consulte "Seguridad de publicación/suscripción" en la página 472
4. En los releases anteriores a IBM MQ for z/OS V6, la comprobación de seguridad se realizaba para MVS.START.STC.CSQ1CHIN. En IBM MQ for z/OS V6 y posterior, se ha añadido el calificador adicional JOBNAME al nombre de recurso. Esto puede causar problemas al iniciar el iniciador de canal.

   Para resolver el problema, sustituya MVS.START.STC. *ssid* CHIN con un perfil para un recurso denominado MVS.START.STC. *ssid* CHIN **.*** o MVS.START.STC. *ssid* CHIN. *ssid* CHIN donde *ssid* es el ID de subsistema del gestor de colas. Para esto es necesaria autorización UPDATE de RACF. Para obtener más detalles, consulte el z/OS documentación del producto correspondiente a *Planificación de operaciones, mandatos MVS, autorizaciones de acceso RACF y nombres de recurso*.

   El mandato START para *ssid*MSTR no incluye el parámetro JOBNAME=. Con fines de coherencia, puede ser conveniente actualizar el perfil de MVS.START.STC.ssidMSTR a MVS.START.STC.ssidMSTR.*.

Tabla 50. Mandatos PCF, perfiles y sus niveles de acceso

Mandato	**Perfil de mandato para MQCMDS**	**Nivel de acceso para MQCMDS**	**Perfil de recurso de mandato para MQADMIN o MXADMIN**	**Nivel de acceso para MQADMIN o MXADMIN**
Hacer copia de seguridad de estructura CF	hlq.BACKUP.CFSTRUCT	CONTROL	No se comprueba	-
Cambiar objeto de información de autenticación	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Cambiar estructura CF	hlq.ALTER.CFSTRUCT	ALTER	No se comprueba	-
Cambiar canal	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Modificar lista de nombres	hlq.ALTER.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Modificar proceso	hlq.ALTER.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Modificar cola	hlq.ALTER.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Cambiar gestor de colas	hlq.ALTER.QMGR	ALTER	No se comprueba	-
Cambiar seguridad	hlq.ALTER.SECURITY	ALTER	No se comprueba	-
Cambiar SMDS	hlq.ALTER.SMDS	ALTER	No se comprueba	-
Cambiar clase de almacenamiento	hlq.ALTER.STGCLASS	ALTER	No se comprueba	-
Cambiar suscripción	hlq.ALTER.SUB	ALTER	No se comprueba	-
Cambiar tema	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Borrar cola	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER

Tabla 50. Mandatos PCF, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
Borrar tema String "1" en la página 230	hlq.CLEAR.TOPICSTR	ALTER	hlq.TOPIC.topic	ALTER
Copiar objeto de información de autenticación	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Copiar estructura CF	hlq.DEFINE.CFSTRUCT	ALTER	No se comprueba	-
Copiar canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Copiar lista de nombres	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Copiar proceso	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Copiar cola	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Copiar suscripción	hlq.DEFINE.SUB	ALTER	No se comprueba	-
Copiar clase de almacenamiento	hlq.DEFINE.STGCLASS	ALTER	No se comprueba	-
Copiar tema	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Crear objeto de información de autenticación	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Crear estructura CF	hlq.DEFINE.CFSTRUCT	ALTER	No se comprueba	-
Crear canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Crear lista de nombres	hlq.DEFINE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Crear proceso	hlq.DEFINE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Crear cola	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Crear clase de almacenamiento	hlq.DEFINE.STGCLASS	ALTER	No se comprueba	-
Crear suscripción	hlq.DEFINE.SUB	ALTER	No se comprueba	-
Crear tema	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Suprimir objeto de información de autenticación	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Suprimir estructura CF	hlq.DELETE.CFSTRUCT	ALTER	No se comprueba	-
Suprimir canal	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Suprimir lista de nombres	hlq.DELETE.NAMELIST	ALTER	hlq.NAMELIST.namelist	ALTER
Suprimir proceso	hlq.DELETE.PROCESS	ALTER	hlq.PROCESS.process	ALTER
Suprimir cola	hlq.DELETE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Suprimir clase de almacenamiento	hlq.DELETE.STGCLASS	ALTER	No se comprueba	-
Suprimir suscripción	hlq.DELETE.SUB	ALTER	No se comprueba	-
Suprimir tema	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER

Tabla 50. Mandatos PCF, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
Consultar archivo	hlq.DISPLAY.ARCHIVE	READ	No se comprueba	-
Consultar objeto de información de autenticación	hlq.DISPLAY.AUTHINFO	READ	No se comprueba	-
Consultar nombres de objeto de información de autenticación	hlq.DISPLAY.AUTHINFO	READ	No se comprueba	-
Consultar estructura CF	hlq.DISPLAY.CFSTRUCT	READ	No se comprueba	-
Consultar nombres de estructura CF	hlq.DISPLAY.CFSTRUCT	READ	No se comprueba	-
Consultar estado de estructura CF	hlq.DISPLAY.CFSTATUS	READ	No se comprueba	-
Consultar canal	hlq.DISPLAY.CHANNEL	READ	No se comprueba	-
Consultar registros de autenticación de canal	hlq.DISPLAY.CHLAUTH	READ	No se comprueba	-
Consultar iniciador de canal	hlq.DISPLAY.CHINIT	READ	No se comprueba	-
Consultar nombres de canal	hlq.DISPLAY.CHANNEL	READ	No se comprueba	-
Consultar estado del canal	hlq.DISPLAY.CHSTATUS	READ	No se comprueba	-
Consultar gestor de colas de clúster	hlq.DISPLAY.CLUSQMGR	READ	No se comprueba	-
Consultar conexión	hlq.DISPLAY.CONNPCF	READ	No se comprueba	-
Consultar grupo	hlq.DISPLAY.GROUP	READ	No se comprueba	-
Consultar registro	hlq.DISPLAY.LOG	READ	No se comprueba	-
Consultar lista de nombres	hlq.DISPLAY.NAMELIST	READ	No se comprueba	-
Consultar nombres de lista de nombres	hlq.DISPLAY.NAMELIST	READ	No se comprueba	-
Consultar proceso	hlq.DISPLAY.PROCESS	READ	No se comprueba	-
Consultar nombres de proceso	hlq.DISPLAY.PROCESS	READ	No se comprueba	-
Consultar estado de publicación/suscripción	hlq.DISPLAY.PUBSUB	READ	No se comprueba	-
Consultar cola	hlq.DISPLAY.QUEUE	READ	No se comprueba	-
Consultar gestor de colas	hlq.DISPLAY.QMGR	READ	No se comprueba	-
Consultar nombres de cola	hlq.DISPLAY.QUEUE	READ	No se comprueba	-
Consultar estado de la cola	hlq.DISPLAY.QSTATUS	READ	No se comprueba	-

Tabla 50. Mandatos PCF, perfiles y sus niveles de acceso (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
Consultar seguridad	hlq.DISPLAY.SECURITY	READ	No se comprueba	-
Consultar SMDS	hlq.DISPLAY.SMDS	READ	No se comprueba	-
Consultar SMDSCONN	hlq.DISPLAY.SMDSCONN	READ	No se comprueba	-
Consultar clase de almacenamiento	hlq.DISPLAY.STGCLASS	READ	No se comprueba	-
Consultar nombres de clase de almacenamiento	hlq.DISPLAY.STGCLASS	READ	No se comprueba	-
Consultar suscripción	hlq.INQUIRE.SUB	READ	No se comprueba	-
Consultar estado de suscripción	hlq.INQUIRE.SBSTATUS	READ	No se comprueba	-
Consultar sistema	hlq.DISPLAY.SYSTEM	READ	No se comprueba	-
Consultar tema	hlq.DISPLAY.TOPIC	READ	No se comprueba	-
Consultar nombres de temas	hlq.DISPLAY.TOPIC	READ	No se comprueba	-
Consultar estado de tema	hlq.DISPLAY.TPSTATUS	READ	No se comprueba	-
Consultar utilización	hlq.DISPLAY.USAGE	READ	No se comprueba	-
Mover cola	hlq.MOVE.QLOCAL	ALTER	hlq.QUEUE.from-queue hlq.QUEUE.to-queue	ALTER
Sondear canal	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Recuperar estructura CF	hlq.RECOVER.CFSTRUCT	CONTROL	No se comprueba	-
Renovar clúster	hlq.REFRESH.CLUSTER	ALTER	No se comprueba	-
Renovar gestor de colas	hlq.REFRESH.QMGR	ALTER	No se comprueba	-
Renovar seguridad	hlq.REFRESH.SECURITY	ALTER	No se comprueba	-
Restablecer estructura CF	hlq.RESET.CFSTRUCT	CONTROL	No se comprueba	-
Restablecer canal	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Restablecer clúster	hlq.RESET.CLUSTER	CONTROL	No se comprueba	-
Restablecer gestor de colas	hlq.RESET.QMGR	CONTROL	No se comprueba	-
Restablecer estadísticas de la cola	hlq.RESET.QSTATS	CONTROL	hlq.QUEUE.queue	CONTROL
Restablecer SMDS	hlq.RESET.SMDS	CONTROL	No se comprueba	-
Resolver canal	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Reanudar gestor de colas	hlq.RESUME.QMGR	CONTROL	No se comprueba	-
Reanudar el clúster de gestores de colas	hlq.RESUME.QMGR	CONTROL	No se comprueba	-

Tabla 50. Mandatos PCF, perfiles y sus niveles de acceso (continuación)				
Mandato	**Perfil de mandato para MQCMDS**	**Nivel de acceso para MQCMDS**	**Perfil de recurso de mandato para MQADMIN o MXADMIN**	**Nivel de acceso para MQADMIN o MXADMIN**
Volver a verificar la seguridad	hlq.RVERIFY.SECURITY	ALTER	No se comprueba	-
Establecer archivo	hlq.SET.ARCHIVE	CONTROL	No se comprueba	-
Establecer registro de autenticación de canal	hlq.SET.CHLAUTH	CONTROL	No se comprueba	-
Establecer registro	hlq.SET.LOG	CONTROL	No se comprueba	-
Establecer sistema	hlq.SET.SYSTEM	CONTROL	No se comprueba	-
Iniciar canal	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Iniciar iniciador de canal	hlq.START.CHINIT	CONTROL	No se comprueba	-
Iniciar escucha de canal	hlq.START.LISTENER	CONTROL	No se comprueba	-
Iniciar conexión SMDS	hlq.START.SMDSCONN	CONTROL	No se comprueba	-
Detener canal	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Detener iniciador de canal	hlq.STOP.CHINIT	CONTROL	No se comprueba	-
Detener escucha de canal	hlq.STOP.LISTENER	CONTROL	No se comprueba	-
Detener conexión SMDS	hlq.STOP.SMDSCONN	CONTROL	No se comprueba	-
Suspender gestor de colas	hlq.SUSPEND.QMGR	CONTROL	No se comprueba	-
Suspender un clúster de gestores de colas	hlq.SUSPEND.QMGR	CONTROL	No se comprueba	-

Notas:

1. El recurso **`hlq.TOPIC.topic`** hace referencia al objeto de tema derivado de TOPICSTR. Para obtener más detalles, consulte "Seguridad de publicación/suscripción" en la página 472

V 9.1.0 Consulte "IBM MQ Console - perfiles de seguridad de mandato necesarios" en la página 230 para obtener detalles de los perfiles PCF de IBM MQ necesarios, al utilizar IBM MQ Console.

z/OS V 9.1.0 *IBM MQ Console - perfiles de seguridad de mandato necesarios*

Las operaciones realizadas en el IBM MQ Console por un usuario en el rol `MQWebAdmin` o `MQWebAdminRO` tienen lugar bajo el contexto de seguridad del ID de usuario de la tarea iniciada del servidor mqweb. Si desea utilizar IBM MQ Console, el ID de usuario de la tarea iniciada del servidor mqweb necesita autorización para emitir determinados mandatos PCF.

Tabla 51 en la página 231 muestra, para cada mandato PCF de IBM MQ, los perfiles de seguridad de mandato necesarios, y el nivel de acceso correspondiente para cada perfil en la clase MQCMDS que necesita IBM MQ Console.

Tabla 51. Mandatos, perfiles y niveles de acceso PCF de IBM MQ Console

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
Cambiar objeto de información de autenticación	hlq.ALTER.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Cambiar canal	hlq.ALTER.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Modificar cola	hlq.ALTER.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Cambiar gestor de colas	hlq.ALTER.QMGR	ALTER	No se comprueba	-
Cambiar tema	hlq.ALTER.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Borrar cola	hlq.CLEAR.QLOCAL	ALTER	hlq.QUEUE.queue	ALTER
Crear objeto de información de autenticación	hlq.DEFINE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Crear canal	hlq.DEFINE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Crear cola	hlq.DEFINE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Crear suscripción	hlq.DEFINE.SUB	ALTER	No se comprueba	-
Crear tema	hlq.DEFINE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Suprimir objeto de información de autenticación	hlq.DELETE.AUTHINFO	ALTER	hlq.AUTHINFO.resourcename	ALTER
Suprimir canal	hlq.DELETE.CHANNEL	ALTER	hlq.CHANNEL.channel	ALTER
Suprimir cola	hlq.DELETE.QUEUE	ALTER	hlq.QUEUE.queue	ALTER
Suprimir suscripción	hlq.DELETE.SUB	ALTER	No se comprueba	-
Suprimir tema	hlq.DELETE.TOPIC	ALTER	hlq.TOPIC.topic	ALTER
Consultar objeto de información de autenticación	hlq.DISPLAY.AUTHINFO	READ	No se comprueba	-
Consultar nombres de objeto de información de autenticación	hlq.DISPLAY.AUTHINFO	READ	No se comprueba	-
Consultar canal	hlq.DISPLAY.CHANNEL	READ	No se comprueba	-
Consultar registros de autenticación de canal	hlq.DISPLAY.CHLAUTH	READ	No se comprueba	-
Consultar iniciador de canal	hlq.DISPLAY.CHINIT	READ	No se comprueba	-
Consultar nombres de canal	hlq.DISPLAY.CHANNEL	READ	No se comprueba	-
Consultar estado del canal	hlq.DISPLAY.CHSTATUS	READ	No se comprueba	-
Consultar cola	hlq.DISPLAY.QUEUE	READ	No se comprueba	-
Consultar gestor de colas	hlq.DISPLAY.QMGR	READ	No se comprueba	-

Tabla 51. Mandatos, perfiles y niveles de acceso PCF de IBM MQ Console (continuación)

Mandato	Perfil de mandato para MQCMDS	Nivel de acceso para MQCMDS	Perfil de recurso de mandato para MQADMIN o MXADMIN	Nivel de acceso para MQADMIN o MXADMIN
Consultar nombres de cola	hlq.DISPLAY.QUEUE	READ	No se comprueba	-
Consultar estado de la cola	hlq.DISPLAY.QSTATUS	READ	No se comprueba	-
Consultar suscripción	hlq.INQUIRE.SUB	READ	No se comprueba	-
Consultar estado de suscripción	hlq.INQUIRE.SBSTATUS	READ	No se comprueba	-
Consultar tema	hlq.DISPLAY.TOPIC	READ	No se comprueba	-
Consultar nombres de temas	hlq.DISPLAY.TOPIC	READ	No se comprueba	-
Consultar estado de tema	hlq.DISPLAY.TPSTATUS	READ	No se comprueba	-
Sondear canal	hlq.PING.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Renovar clúster	hlq.REFRESH.CLUSTER	ALTER	No se comprueba	-
Renovar seguridad	hlq.REFRESH.SECURITY	ALTER	No se comprueba	-
Restablecer canal	hlq.RESET.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Resolver canal	hlq.RESOLVE.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Establecer registro de autenticación de canal	hlq.SET.CHLAUTH	CONTROL	No se comprueba	-
Iniciar canal	hlq.START.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL
Detener canal	hlq.STOP.CHANNEL	CONTROL	hlq.CHANNEL.channel	CONTROL

z/OS *Perfiles para la seguridad de recursos de mandatos*

Si no ha definido el perfil de conmutador de seguridad de recursos de mandato, porque desea comprobar la seguridad de los recursos asociados a mandatos, debe añadir perfiles de recurso para cada recurso a la clase adecuada. Los mismos perfiles de seguridad controlan tanto mandatos MQSC como PCF.

Si no ha definido el perfil de conmutador de seguridad de recursos de mandato, `hlq.NO.CMD.RESC.CHECKS`, porque desea comprobar la seguridad de los recursos asociados a mandatos, debe:

- Añadir un perfil de recurso a la clase **MQADMIN**, si se utilizan perfiles en mayúsculas, para cada recurso.
- Añadir un perfil de recurso a la clase **MXADMIN**, si se utilizan perfiles que combinan mayúsculas y minúsculas, para cada recurso.

Los mismos perfiles de seguridad controlan tanto mandatos MQSC como PCF.

Los perfiles para la comprobación de seguridad de recursos de mandatos adoptan la forma:

```
hlq.type.resourcename
```

donde `hlq` puede ser `qmgr-name` (nombre de gestor de colas) o `qsg-name` (nombre de grupo de compartición de colas).

Un perfil que lleve como prefijo el nombre del gestor de colas controla el acceso a los recursos asociados a mandatos de dicho gestor de colas. Un perfil que lleve como prefijo el nombre del grupo de compartición de colas controla el acceso a los recursos asociados a mandatos en todos los gestores

de colas en el grupo de compartición de colas. Este acceso se puede sustituir en un gestor de colas individual definiendo un perfil de nivel de gestor de colas para dicho recurso de mandato en es gestor de colas.

Si el gestor de colas es miembro de un grupo de compartición de colas y se está utilizando seguridad a nivel tanto de gestor de colas como de grupo de compartición de colas, IBM MQ busca primero un perfil que tenga como prefijo el nombre del gestor de colas. Si no encuentra ninguno, busca un perfil que tenga como prefijo el nombre del grupo de compartición de colas.

Por ejemplo, el nombre del perfil RACF para la comprobación de seguridad de recursos de mandatos para la cola modelo CREDIT.WORTHY en el subsistema CSQ1 es:

```
CSQ1.QUEUE.CREDIT.WORTHY
```

Puesto que los perfiles para todos los tipos de recurso de mandato se almacenan en la clase MQADMIN, la parte "type" del nombre de perfil es necesaria en el perfil para distinguir entre recursos de tipos diferentes que tengan el mismo nombre. La parte "type" del nombre de perfil puede ser CHANNEL, QUEUE, TOPIC, PROCESS o NAMELIST. Por ejemplo, un usuario puede estar autorizado a definir hlq.QUEUE.PAYROLL.ONE, pero no estar autorizado a definir hlq.PROCESS.PAYROLL.ONE

Si el tipo de recurso es una cola y el perfil es un perfil a nivel de grupo de compartición de colas, éste controla el acceso a una o varias colas locales dentro del grupo de compartición de colas o el acceso a una sola cola compartida de cualquier gestor de colas del grupo de compartición de colas.

z/OS La tabla Mandatos MQSC, perfiles y sus niveles de acceso muestra, para cada mandato MQSC de IBM MQ, los perfiles necesarios para llevar a cabo la comprobación de seguridad de mandatos, y el nivel de acceso correspondiente para cada perfil de la clase MQCMDS.

z/OS La tabla Mandatos PCF, perfiles y sus niveles de acceso muestra, para cada mandato PCF de IBM MQ, los perfiles necesarios para llevar a cabo la comprobación de seguridad de mandatos, y el nivel de acceso correspondiente para cada perfil de la clase MQCMDS.

z/OS *Comprobación de seguridad de recursos de mandatos para colas alias y colas remotas*
La cola alias y las colas remotas proporcionan ambas direccionamiento indirecto a otra cola. Hay cuestiones adicionales a tener en cuenta al considerar la comprobación de seguridad para estas colas.

Colas alias

Cuando se define una cola alias, las comprobaciones de seguridad de recursos de mandatos sólo se realizan para el nombre de la cola alias, no para el nombre de la cola de destino en que se resuelve el alias.

Las colas alias se pueden resolver en colas locales y colas remotas. Si no desea permitir a los usuarios el acceso a ciertas colas locales o remotas, debe realizar las acciones siguientes:

1. No permitir a los usuarios el acceso a estas colas locales y remotas.
2. Impedir que los usuarios puedan definir alias para estas colas. Es decir, impedirles que puedan emitir mandatos DEFINE QALIAS y ALTER QALIAS.

Colas remotas

Cuando se define una cola remota, las comprobaciones de seguridad de recursos de mandatos sólo se realizan para el nombre de la cola remota. No se realizan comprobaciones para los nombres de las colas especificadas en los atributos RNAME o XMITQ de la definición de objeto de cola remota.

z/OS El perfil de seguridad RESLEVEL

Puede definir un perfil especial en la clase MQADMIN o MXADMIN para controlar el número de identificadores (ID) de usuario que se comprueban para la seguridad de recursos de la API. Este perfil

se denomina el perfil RESLEVEL. El modo en que este perfil afecta a la seguridad de recursos de la API depende de cómo se accede a IBM MQ.

Cuando una aplicación intenta conectarse a IBM MQ, IBM MQ comprueba el acceso que el ID de usuario asociado a la conexión tiene a un perfil en la clase MQADMIN o MXADMIN llamado:

```
hlq.RESLEVEL
```

Donde `hlq` puede ser `ssid` (ID del subsistema) o `qsg` (ID del grupo de compartición de colas).

Los ID de usuario asociados a cada tipo de conexión son:

- El ID de usuario de la tarea de conexión para las conexiones por lotes
- El ID de usuario del espacio de direcciones del CICS para conexiones CICS
- El ID de usuario del espacio de direcciones de la región IMSpara conexiones de IMS
- El ID de usuario del espacio de direcciones del iniciador de canal para las conexiones del iniciador de canal

Atención: RESLEVEL es una opción muy potente; puede hacer que se pasen por alto todas las comprobaciones de seguridad de recursos para una determinada conexión.

Si no tiene un perfil RESLEVEL definido, asegúrese de que ningún otro perfil de la clase MQADMIN coincida con hlq.RESLEVEL. Por ejemplo, si tiene un perfil en MQADMIN llamado hlq. * * y ningún perfil hlq.RESLEVEL , tenga cuidado con las consecuencias del hlq. * * porque se utiliza para la comprobación RESLEVEL.

Defina un perfil hlq.RESLEVEL y establezca el UACC en NONE, en vez de no tener perfil RESLEVEL alguno. Tenga el menor número de usuarios o grupos en la lista de acceso que sea posible. Para obtener información detallada sobre cómo auditar el acceso RESLEVEL, consulte “Consideraciones de auditoría en z/OS” en la página 261.

Si está utilizando solamente seguridad a nivel de gestor de colas, IBM MQ realiza comprobaciones RESLEVEL para el perfil `qmgr-name.RESLEVEL`. Si está utilizando solamente seguridad a nivel de grupo de compartición de colas, IBM MQ realiza comprobaciones RESLEVEL para el perfil `qsg-name.RESLEVEL`. Si está utilizando una combinación de seguridad a nivel de gestor de colas y de grupo de compartición de colas, IBM MQ comprueba primero la existencia de un perfil RESLEVEL a nivel de gestor de colas. Si no encuentra ninguno, busca un perfil RESLEVEL a nivel de grupo de compartición de colas.

Si no puede encontrar un perfil RESLEVEL, IBM MQ permite la comprobación del ID de trabajo y tarea (o de usuario alternativo) para una conexión CICS o IMS. Para una conexión por lotes, IBM MQ permite la comprobación del ID de usuario del trabajo (o alternativo). Para el iniciador de canal, IBM MQ permite la comprobación del ID de usuario de canal y el ID de usuario de MCA (o alternativo).

Si hay un perfil RESLEVEL, el nivel de comprobación depende del entorno y el nivel de acceso para el perfil.

Recuerde que si el gestor de colas es miembro de un grupo de compartición de colas y no define este perfil a nivel de gestor de colas, puede haber uno definido a nivel de grupo de compartición de colas que afectará al nivel de comprobación. Para activar la comprobación de dos ID de usuario, defina un perfil RESLEVEL (con un prefijo que es el nombre del gestor de colas del nombre del grupo de compartición de colas) con un UACC(NONE) y asegúrese de que los usuarios relevantes no tengan acceso otorgado en este perfil.

Cuando considere el acceso que el ID de usuario del iniciador del canal tiene a RESLEVEL, recuerde que la conexión establecida por el iniciador de canal es también la conexión utilizada por los canales. Un valor que hace que se pasen por alto todas las comprobaciones de seguridad de recursos para el ID de usuario del iniciador del canal omite de hecho las comprobaciones de seguridad para todos los canales. Si el ID de usuario del iniciador del canal es cualquiera que no sea NONE, entonces sólo se comprueba el acceso de un ID de usuario (para un nivel de acceso READ o UPDATE) o de ningún ID de usuario (para un nivel de acceso CONTROL o ALTER). si otorga al ID de usuario del iniciador de canal un nivel de acceso distinto

de NONE a RESLEVEL, asegúrese de que comprende el efecto de este valor en las comprobaciones de seguridad realizadas para los canales.

La utilización del perfil RESLEVEL significa que no se toman registros de auditoría de seguridad normales. Por ejemplo, si pone UAUDIT en un usuario, no se realiza la auditoría del acceso al perfil hlq.RESLEVEL en MQADMIN.

Si utiliza la opción WARNING de RACF en el perfil hlq.RESLEVEL, no se generan mensajes de aviso RACF para los perfiles de la clase RESLEVEL.

Las comprobaciones de seguridad para los mensajes de informes como los COD los controla el perfil RESLEVEL asociado a la aplicación de origen. Por ejemplo, si un ID de usuario de trabajo por lotes tiene autorización de CONTROL o ALTER para un perfil RESLEVEL, entonces todas las comprobaciones de recursos realizadas por el trabajo por lotes se omiten , incluida la comprobación de seguridad de los mensajes de informes.

Si cambia el perfil RESLEVEL, los usuarios deben desconectarse y conectarse de nuevo para que el cambio tenga efecto. (Esto incluye la detención y reinicio del iniciador de canal si se cambia el acceso que el ID de usuario del espacio de direcciones de gestión de colas distribuidas tiene al perfil RESLEVEL.)

Para desactivar la auditoría de RESLEVEL, utilice el parámetro del sistema RESAUDIT.

z/OS *RESLEVEL y las conexiones por lotes*

De forma predeterminada, cuando se accede a un recurso de IBM MQ a través de conexiones por lotes o de tipo por lotes, el usuario debe estar autorizado a acceder a dicho recurso para la operación específica. Puede eludir la comprobación de seguridad estableciendo una definición RESLEVEL adecuada.

Que el usuario se compruebe o no depende del ID de usuario utilizado durante la conexión, ya que se utiliza el mismo ID de usuario para la comprobación de conexión.

Por ejemplo, puede configurar RESLEVEL de manera que cuando un usuario de confianza acceda a determinados recursos a través de una conexión por lotes, no se realice ninguna comprobación de seguridad de recursos de la API; pero cuando un usuario que no sea de confianza intente acceder los mismos recursos, se lleven a cabo las comprobaciones de seguridad habituales. Sólo debería configurar la comprobación RESLEVEL para eludir las comprobaciones de seguridad de recursos de la API cuando tenga la suficiente confianza en el usuario y los programas ejecutados por dicho usuario.

La tabla siguiente muestra las comprobaciones que se realizan para las conexiones por lotes.

Tabla 52. Comprobaciones realizadas en diferentes niveles de acceso RACF para conexiones por lotes

Nivel de acceso de RACF	**Nivel de comprobación**
NINGUNO	Se realizan comprobaciones de recursos
READ	Se realizan comprobaciones de recursos
UPDATE	Se realizan comprobaciones de recursos
CONTROL	No hay ninguna comprobación.
ALTER	No hay ninguna comprobación.

z/OS *RESLEVEL y las funciones del sistema*

La aplicación de RESLEVEL a los paneles de operaciones y paneles de control, y a CSQUTIL.

Los paneles de operaciones y paneles de control y el programa de utilidad CSQUTIL son aplicaciones de tipo por lotes que realizan peticiones al servidor de mandatos del gestor de colas y, por lo tanto, están sujetos a las consideraciones descritas en “RESLEVEL y las conexiones por lotes” en la página 235. Puede utilizar RESLEVEL para omitir la comprobación de seguridad para SYSTEM.COMMAND.INPUT y SYSTEM.COMMAND.REPLY.MODEL que utilizan, pero no para las colas dinámicas SYSTEM.CSQXCMD. *, SYSTEM.CSQOREXX.*, y SYSTEM.CSQUTIL. *.

El servidor de mandatos es una parte integral del gestor de colas, por lo que no tiene comprobación de conexión o RESLEVEL asociada a él. Por lo tanto, para mantener la seguridad, el servidor de mandatos debe confirmar que el ID de usuario de la aplicación que realiza la solicitud tiene autorización para abrir la cola que se utiliza para las respuestas. Para los paneles de operaciones y paneles de control, esta cola es SYSTEM.CSQOREXX.*. Para CSQUTIL, es SYSTEM.CSQUTIL.*. Los usuarios deben contar con autorización para utilizar estas colas, como se describe en “Seguridad de colas del sistema” en la página 207, además de contar con la autorización RESLEVEL que se les proporciona.

Para otras aplicaciones que utilicen el servidor de mandatos, es la cola que éstas designen como su cola de respuestas. Estas otras aplicaciones pueden llevar engañosamente al servidor de mandatos a colocar mensajes en colas no autorizadas, pasando (en el contexto de mensaje) un ID de usuario más fiable que el suyo propio al servidor de mandatos. Para impedir esto, utilice un perfil CONTEXT para proteger el contexto de identidad de los mensajes colocados en SYSTEM.COMMAND.INPUT.

z/OS *RESLEVEL y las conexiones CICS*

De forma predeterminada, cuando se realiza una comprobación de seguridad de recursos de la API en una conexión CICS, se comprueban dos ID de usuario. Puede cambiar los ID de usuario que se comprueban mediante la configuración de un perfil RESLEVEL.

El primer ID de usuario que se comprueba es el del espacio de direcciones CICS. Este es el ID de usuario existente en la tarjeta de trabajo del trabajo CICS, o el ID de usuario asignado a la tarea iniciada CICS por la clase STARTED de z/OS o la tabla de procedimientos iniciados. (No es el ID de usuario DFLTUSER de CICS.)

El segundo ID de usuario que se comprueba es el ID de usuario asociado a la transacción CICS.

Si uno de estos ID de usuario no tiene acceso al recurso, la solicitud falla con un código de terminación MQRC_NOT_AUTHORIZED. Tanto el ID de usuario del espacio de direcciones CICS como el ID de usuario de la persona que ejecuta la transacción CICS deben tener acceso al recurso en el nivel correcto.

Cómo puede afectar RESLEVEL a las comprobaciones realizadas

Dependiendo de cómo configure el perfil RESLEVEL, puede cambiar qué identificadores (ID) de usuario se comprueban cuando se solicita acceso a un recurso. Consulte Tabla 53 en la página 236 para obtener más información.

Los ID de usuario que se comprueban dependen del ID de usuario utilizado durante la conexión, es decir, el ID de usuario del espacio de direcciones CICS. Este control le permite eludir la comprobación de seguridad de recursos de la API para solicitudes IBM MQ procedentes de un sistema (por ejemplo, un sistema de prueba TESTCICS,) pero implementarla para otro (por ejemplo, un sistema de producción, PRODCICS).

Nota: Si configura el ID de usuario del espacio de direcciones CICS con el atributo "trusted" en la clase STARTED o la tabla de procedimientos iniciados RACF ICHRIN03, esto altera temporalmente todas las comprobaciones de ID de usuario para el espacio de direcciones CICS establecidas por el perfil RESLEVEL para el gestor de colas (es decir, el gestor de colas no realiza las comprobaciones de seguridad para el espacio de direcciones CICS). Para más información, consulte *CICS Transaction Server for z/OS V3.2 RACF Security Guide*.

La tabla siguiente muestra las comprobaciones que se realizan para las conexiones CICS.

Tabla 53. Comprobaciones realizadas en diferentes niveles de acceso RACF para conexiones CICS

Nivel de acceso de RACF	**Nivel de comprobación**
NINGUNO	IBM MQ comprueba el ID del usuario del espacio de direcciones de CICS y el ID de usuario de la transacción.
READ	IBM MQ solo comprueba el ID de usuario del espacio de direcciones de CICS.

Tabla 53. Comprobaciones realizadas en diferentes niveles de acceso RACF para conexiones CICS (continuación)	
Nivel de acceso de RACF	**Nivel de comprobación**
UPDATE	Si la transacción se ha definido en CICS con RESSEC(YES), IBM MQ comprueba el ID de usuario del espacio de direcciones de CICS y el ID de usuario de la transacción.
UPDATE	Si la transacción se ha definido en CICS con RESSEC(NO), IBM MQ solo comprueba el ID de usuario del espacio de direcciones de CICS.
CONTROL o ALTER	IBM MQ no comprueba ningún ID de usuario.

z/OS *RESLEVEL y las conexiones IMS*

De forma predeterminada, cuando se realiza una comprobación de seguridad de recursos de la API para una conexión IMS, se comprueban dos ID de usuario. Puede cambiar los ID de usuario que se comprueban mediante la configuración de un perfil RESLEVEL.

De forma predeterminada, cuando se realiza una comprobación de seguridad de recursos de la API para una conexión IMS, se comprueban dos ID de usuario para ver si se permite el acceso al recurso.

El primer ID de usuario que se comprueba es el del espacio de direcciones de la región IMS. Éste se obtiene del campo USER de la tarjeta de trabajo o del ID de usuario asignado a la región por la clase STARTED de z/OS o la tabla de procedimientos iniciados (SPT).

El segundo ID de usuario que se comprueba está asociado al trabajo que se realiza en la región dependiente. Se determina en función del tipo de la región dependiente, como se muestra en Cómo se determina el segundo ID de usuario para la conexión IMS(tm).

Si el primero o segundo ID de usuario IMS no tiene acceso al recurso, la solicitud falla con un código de terminación MQRC_NOT_AUTHORIZED.

El valor de los perfiles RESLEVEL de IBM MQ no puede alterar el ID de usuario bajo el que se planifican las transacciones IMS desde el programa supervisor desencadenante CSQQTRMN de MQ-IMS proporcionado por IBM. Este ID de usuario es el PSBNAME de dicho supervisor desencadenante, que es CSQQTRMN de forma predeterminada.

Cómo puede afectar RESLEVEL a las comprobaciones realizadas

Dependiendo de cómo configure el perfil RESLEVEL, puede cambiar qué identificadores (ID) de usuario se comprueban cuando se solicita acceso a un recurso. Las comprobaciones posibles son:

- Comprobar el ID de usuario del espacio de direcciones de la región IMS y el segundo ID de usuario o ID de usuario alternativo.
- Comprobar sólo el ID de usuario del espacio de direcciones de la región IMS.
- No comprobar ningún ID de usuario.

La tabla siguiente muestra las comprobaciones que se realizan para las conexiones IMS.

Tabla 54. Comprobaciones realizadas en diferentes niveles de acceso RACF para conexiones IMS	
Nivel de acceso de RACF	**Nivel de comprobación**
NINGUNO	Comprobar el ID de usuario del espacio de direcciones de IMS y el segundo ID de usuario o ID de usuario alternativo de IMS.
READ	Comprobar el ID de usuario de espacio de direcciones de IMS
UPDATE	Comprobar el ID de usuario de espacio de direcciones de IMS
CONTROL	No hay ninguna comprobación.
ALTER	No hay ninguna comprobación.

z/OS *RESLEVEL y la conexión del iniciador de canal*

De forma predeterminada, cuando el iniciador de canal realiza una comprobación de seguridad de recursos de la API, se comprueban dos ID de usuario. Puede cambiar los ID de usuario que se comprueban mediante la configuración de un perfil RESLEVEL.

De forma predeterminada, cuando el iniciador de canal realiza una comprobación de seguridad de recursos de la API, se comprueban dos ID de usuario para ver si se permite el acceso al recurso.

Los ID de usuario que se comprueban pueden ser el especificado por el atributo de canal MCAUSER, el que se ha recibido de la red, el del espacio de direcciones del iniciador de canal o el ID de usuario alternativo para el descriptor de mensaje. Cuáles son los ID de usuario que se comprueban depende del protocolo de comunicación que esté utilizando y del valor del atributo de canal PUTAUT. Consulte "Identificadores de usuario utilizados por el iniciador de canal" en la página 243 para obtener más información.

Si uno de estos ID de usuario no tiene acceso al recurso, la solicitud falla con un código de terminación MQRC_NOT_AUTHORIZED.

Cómo puede afectar RESLEVEL a las comprobaciones realizadas

Dependiendo de cómo configure el perfil RESLEVEL, puede cambiar qué identificadores (ID) de usuario se comprueban cuando se solicita acceso a un recurso, y cuántos se comprueban.

La tabla siguiente muestra las comprobaciones que se realizan para la conexión del iniciador de canal y para todos los canales, ya que estos utilizan esta conexión.

Tabla 55. Comprobaciones realizadas en diferentes niveles de acceso RACF para conexiones de iniciador de canal

Nivel de acceso de RACF	**Nivel de comprobación**
NINGUNO	Comprobar dos ID de usuario.
READ	Comprobar un ID de usuario.
UPDATE	Comprobar un ID de usuario.
CONTROL	No hay ninguna comprobación.
ALTER	No hay ninguna comprobación.
Nota: Consulte "Identificadores de usuario utilizados por el iniciador de canal" en la página 243 para obtener una definición de los ID de usuarios comprobados.	

z/OS *RESLEVEL y la transferencia a colas entre grupos*

De forma predeterminada, cuando el agente de transferencia a colas entre grupos realiza una comprobación de seguridad de recursos de la API, se comprueban dos ID de usuario para ver si se permite el acceso al recurso. Puede cambiar los ID de usuario que se comprueban mediante la configuración de un perfil RESLEVEL.

Los ID de usuario comprobados pueden ser el ID de usuario determinado por el atributo IGQUSER del gestor de colas receptor, el ID de usuario del gestor de colas dentro del grupo de compartición de colas que ha colocado el mensaje en SYSTEM.QSG.TRANSMIT.QUEUE, o el ID de usuario alternativo especificado en el campo *UserIdentifier* del descriptor de mensaje del mensaje. Consulte "Identificadores de usuario utilizados por el agente de transferencia a colas entre grupos" en la página 248 para obtener más información.

Debido a que el agente de transferencia a colas entre grupos es una tarea interna del gestor de colas, no emite una solicitud de conexión explícita y se ejecuta bajo el ID de usuario del gestor de colas. El agente de transferencia a colas entre grupos se inicia durante la inicialización del gestor de colas. Durante la

inicialización del agente de transferencia a colas entre grupos, IBM MQ comprueba el acceso que el ID de usuario asociado al gestor de colas tiene a un perfil de la clase MQADMIN llamado:

```
hlq.RESLEVEL
```

Esta comprobación se realiza siempre, a menos que se haya establecido el conmutador hlq.NO.SUBSYS.SECURITY.

Si no hay ningún perfil RESLEVEL, IBM MQ permite la comprobación de dos ID de usuario. Si hay un perfil RESLEVEL, el nivel de comprobación depende del nivel de acceso otorgado al ID de usuario del gestor de colas para el perfil. Comprobaciones realizadas en diferentes niveles de acceso RACF(r) para el agente de transferencia a colas entre grupos muestra las comprobaciones que se realizan para el agente de transferencia a colas entre grupos.

Tabla 56. Comprobaciones realizadas en diferentes niveles de acceso RACF para el agente de transferencia a colas entre grupos

Nivel de acceso de RACF	**Nivel de comprobación**
NINGUNO	Comprobar dos ID de usuario.
READ	Comprobar un ID de usuario.
UPDATE	Comprobar un ID de usuario.
CONTROL	No hay ninguna comprobación.
ALTER	No hay ninguna comprobación.
Nota: Consulte “Identificadores de usuario utilizados por el agente de transferencia a colas entre grupos” en la página 248 para obtener una definición de los ID de usuarios comprobados.	

Si se cambian los permisos otorgados al perfil RESLEVEL para el ID de usuario del gestor de colas, el agente de transferencia a colas entre grupos debe detenerse y reiniciarse para captar los nuevos permisos. Puesto que no hay ninguna manera de detener y reiniciar independientemente el agente de transferencia a colas entre grupos, deberá detener y reiniciar el gestor de colas para lograr esto.

z/OS *RESLEVEL y los ID de usuario que se comprueban*

Un ejemplo de cómo definir un perfil RESLEVEL y otorgar acceso al mismo.

Comprobación de ID de usuario en el nombre de perfil para conexiones por lotes hasta Los ID de usuario que se comprueban con el nombre de perfil para canales de conexión con el servidor LU 6.2 y TCP/IP muestran cómo RESLEVEL influye a la hora de determinar qué identificadores (ID) de usuario se comprueban para diferentes peticiones MQI.

Por ejemplo, tiene un gestor de colas llamado QM66 con los siguientes requisitos:

- El usuario WS21B va a estar exento de la seguridad de recursos.
- La tarea iniciada de CICS WXNCICS que se ejecuta bajo el ID de usuario de espacio de direcciones CICSWXN consiste en realizar la comprobación de recursos completa sólo para transacciones definidas con RESSEC(YES).

Para definir el perfil RESLEVEL adecuado, emita el siguiente mandato RACF:

```
RDEFINE MQADMIN QM66.RESLEVEL UACC(NONE)
```

A continuación, otorgue a los usuarios acceso a este perfil, mediante los mandatos siguientes:

```
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(WS21B) ACCESS(CONTROL)
PERMIT QM66.RESLEVEL CLASS(MQADMIN) ID(CICSWXN) ACCESS(UPDATE)
```

Si realiza estos cambios mientras los ID de usuario están conectados al gestor de colas QM66, los usuarios deben desconectarse y conectarse de nuevo para que el cambio tenga lugar.

Si la seguridad de subsistema no está activa cuando un usuario se conecta, pero se activa mientras este usuario sigue estando conectado, se aplica al usuario la comprobación de seguridad de recursos completa. El usuario debe volver a conectarse para obtener el proceso RESLEVEL correcto.

z/OS ID de usuario para la comprobación de seguridad en z/OS

IBM MQ inicia comprobaciones de seguridad basándose en los ID de usuario asociados a usuarios, terminales, aplicaciones y otros recursos. En esta colección de temas se tratan los ID de usuario que se utilizan para cada tipo de comprobación de seguridad.

z/OS *Identificadores de usuario para la seguridad de conexión*

El ID de usuario que se utiliza para la seguridad de conexión depende del tipo de conexión.

Tipo de conexión	Contenido del ID de usuario
Conexión por lotes	El ID de usuario de la tarea de conexión. Por ejemplo: • El ID de usuario TSO • El ID de usuario asignado a un trabajo por lotes mediante el parámetro JCL USER • El ID de usuario asignado a una tarea iniciada mediante la clase STARTED o la tabla de procedimientos iniciados
CICS conexión	El ID de usuario del espacio de direcciones de CICS.
IMS conexión	El ID de usuario del espacio de direcciones de la región IMS.
Conexión del iniciador de canal	El ID de usuario del espacio de direcciones del iniciador de canal.

z/OS *ID de usuario para la seguridad de mandatos o de recursos de mandatos*

El ID de usuario utilizado para la seguridad de mandatos o la seguridad de recursos de mandatos depende de dónde se emite el mandato.

Emitido desde...	Contenido del ID de usuario
CSQINP1, CSQINP2 o CSQINPT	No se realiza ninguna comprobación.
Cola de entrada de mandatos del sistema	El ID de usuario que se encuentra en el *UserIdentifier* del descriptor de mensaje del mensaje que contiene el mandato. Si el mensaje no contiene un *UserIdentifier*, se pasa un ID de usuario de espacios en blanco al gestor de seguridad.
Consola	El ID de usuario que ha iniciado la sesión en la consola. Si no se ha iniciado la sesión en la consola, el ID de usuario predeterminado establecido mediante el parámetro del sistema CMDUSER en CSQ6SYSP. Para emitir mandatos desde una consola, la consola debe tener el atributo z/OS SYS AUTHORITY.
Consola SDSF/TSO	ID de usuario TSO o del trabajo.

Emitido desde...	Contenido del ID de usuario
Paneles de operaciones y paneles de control	ID de usuario TSO. Si va a utilizar los paneles de operaciones y paneles de control, debe tener la autorización adecuada para emitir los mandatos correspondientes a las acciones que elija. Además, debe tener acceso de lectura (READ) a todos los hlq.DISPLAY. Perfiles *object* en la clase MQCMDS porque los paneles utilizan los diversos mandatos DISPLAY para recopilar la información que presentan.
MGCRE	Si MGCRE se utiliza con UTOKEN, el ID de usuario en la UTOKEN. Si MGCRE se emite sin la UTOKEN, se utiliza el ID de usuario TSO o el ID de usuario del trabajo.
CSQ0UTIL	ID de usuario del trabajo.
CSQUTIL	ID de usuario del trabajo.
CSQINPX	ID de usuario del espacio de direcciones del iniciador de canal.

z/OS *Identificadores de usuario para la seguridad de recursos (MQOPEN, MQSUB y MQPUT1)*

Esta información muestra el contenido de los ID de usuario para los ID de usuario normal y alternativo para cada tipo de conexión. El número de comprobaciones se define mediante el perfil RESLEVEL. El ID de usuario que se comprueba es el que se utiliza para las llamadas **MQOPEN**, **MQSUB** o **MQPUT1**.

Nota: Todos los campos de ID de usuario se comprueban exactamente como se reciben. No se realiza ninguna conversión y, por ejemplo, tres campos de ID de usuario que contengan "Bob", "BOB" y "bob" no son equivalentes.

z/OS *Identificadores de usuario que se comprueban para las conexiones por lotes*

El ID de usuario que se comprueba para una conexión por lotes depende de cómo se ejecuta la tarea y de si se ha especificado un ID de usuario alternativo.

Tabla 57. Comprobación de ID de usuario en el nombre de perfil para conexiones por lotes

¿Se ha especificado un ID de usuario alternativo en la apertura?	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
No	-	JOB	JOB
Sí	JOB	JOB	ALT

Clave:

ALT
ID de usuario alternativo.

JOB

- El ID de usuario de un inicio de sesión TSO o USS.
- El ID de usuario asignado a un trabajo por lotes.
- El ID de usuario asignado a una tarea iniciada mediante la clase STARTED o la tabla de procedimientos iniciados.
- El ID de usuario asociado al procedimiento almacenado Db2 que se ejecuta

Un trabajo por lotes está realizando una MQPUT1 a una cola llamada Q1 con RESLEVEL establecido en READ y la comprobación de usuario alternativo desactivada.

Comprobaciones realizadas en diferentes niveles de acceso RACF(r) para conexiones por lotes y Comprobación de ID de usuario en el nombre de perfil para conexiones por lotes muestran que el ID de usuario del trabajo se comprueba con el perfil hlq.Q1.

z/OS *Identificadores de usuario que se comprueban para las conexiones CICS*

Los ID de usuario que se comprueban para las conexiones CICS dependen de si se va a realizar una o dos comprobaciones, y de si se especifica un ID de usuario alternativo.

Tabla 58. Comprobación de ID de usuario en el nombre de perfil para los ID de usuario de tipo CICS

¿Se ha especificado un ID de usuario alternativo en la apertura?	**Perfil hlq.ALTERNATE.USER.userid**	**Perfil hlq.CONTEXT.queuename**	**Perfil hlq.resourcename**
No, 1 comprobación	-	ADS	ADS
No, 2 comprobaciones	-	ADS+TXN	ADS+TXN
Sí, 1 comprobación	ADS	ADS	ADS
Sí, 2 comprobaciones	ADS+TXN	ADS+TXN	ADS+ALT

Clave:

ALT
ID de usuario alternativo

ADS
El ID de usuario asociado al trabajo por lotes CICS o, si CICS se ejecuta como una tarea iniciada, mediante la clase STARTED o la tabla de procedimientos iniciados.

TXN
El ID de usuario asociado a la transacción CICS. Normalmente, éste el ID de usuario del usuario de terminal que inició la transacción. Puede ser el DFLTUSER de CICS, un terminal de seguridad PRESET, o un usuario que ha iniciado la sesión manualmente.

Determine los ID de usuario que se comprueban para las siguientes condiciones:

- El nivel de acceso RACF al perfil RESLEVEL, para un ID de usuario del espacio de direcciones CICS, está establecido en NONE.
- Se realiza una llamada MQOPEN para una cola con MQOO_OUTPUT y MQOO_PASS_IDENTITY_CONTEXT.

En primer lugar, vea cuántos ID de usuario CICS se comprueban basados en el acceso del ID de usuario del espacio de direcciones CICS al perfil RESLEVEL. En la Tabla 53 en la página 236 en el tema "RESLEVEL y las conexiones CICS" en la página 236, se comprueban dos ID de usuario si el perfil RESLEVEL está establecido en NONE. A continuación, en la Tabla 58 en la página 242, se llevan a cabo estas comprobaciones:

- El perfil hlq.ALTERNATE.USER.userid no se comprueba.
- El perfil hlq.CONTEXT.queuename se comprueba tanto con el ID de usuario del espacio de direcciones CICS como con el ID de usuario de la transacción CICS.
- El perfil hlq.resourcename se comprueba tanto con el ID de usuario del espacio de direcciones CICS como con el ID de usuario de la transacción CICS.

Esto significa que se realizan cuatro comprobaciones de seguridad para esta llamada MQOPEN.

z/OS *Identificadores de usuario que se comprueban para las conexiones IMS*

Los ID de usuario que se comprueban para las conexiones IMS dependen de si se va a realizar una o dos comprobaciones, y de si se especifica un ID de usuario alternativo. Si se comprueba un segundo ID

de usuario, éste depende del tipo de región dependiente y de qué identificadores (ID) de usuario están disponibles.

Tabla 59. Comprobación de ID de usuario en el nombre de perfil para los ID de usuario de tipo IMS

¿Se ha especificado un ID de usuario alternativo en la apertura?	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
No, 1 comprobación	-	REG	REG
No, 2 comprobaciones	-	REG+SEC	REG+SEC
Sí, 1 comprobación	REG	REG	REG
Sí, 2 comprobaciones	REG+SEC	REG+SEC	REG+ALT

Clave:

ALT
ID de usuario alternativo.

REG
El ID de usuario normalmente se establece mediante la clase STARTED o la tabla de procedimientos iniciados o, si IMS se está ejecutando, desde un trabajo sometido, mediante el parámetro JCL USER.

SEC
El segundo ID de usuario está asociado al trabajo que se realiza en una región dependiente. Se determina de acuerdo con la Tabla 60 en la página 243.

Tabla 60. Cómo se determina el segundo ID de usuario para la conexión IMS

Tipos de región dependiente	Jerarquía para determinar el segundo ID de usuario
• BMP dirigido por mensajes y GET UNIQUE satisfactorio emitido. • IFP y GET UNIQUE emitido. • MPP.	El ID de usuario asociado a la transacción IMS si el usuario ha iniciado la sesión. Nombre de LTERM, si está disponible. PSBNAME.
• BMP dirigido por mensajes y GET UNIQUE satisfactorio no emitido. • BMP no dirigido por mensajes. • IFP y GET UNIQUE no emitido.	El ID de usuario asociado al espacio de direcciones de la región dependiente IMS si éste no es todo espacios en blanco o todo ceros. PSBNAME.

z/OS *Identificadores de usuario utilizados por el iniciador de canal*

Esta colección de temas describe los ID de usuario utilizados y que se comprueban para canales de recepción y para solicitudes MQI de cliente emitidas a través de canales de conexión con el servidor. Se proporciona información para TCP/IP y para LU6.2

Puede utilizar el parámetro PUTAUT de la definición de canal receptor para determinar el tipo de comprobación de seguridad utilizado. Para conseguir una comprobación de seguridad coherente en toda la red de IBM MQ, puede utilizar las opciones ONLYMCA y ALTMCA.

Puede utilizar el mandato DISPLAY CHSTATUS para determinar el identificador de usuario utilizado por el MCA.

z/OS *Canales receptores que utilizan TCP/IP*

Los ID de usuario que se comprueban dependen de la opción PUTAUT del canal y de si se va a realizar una o dos comprobaciones.

Tabla 61. Los ID de usuario que se comprueban con el nombre de perfil para canales TCP/IP

Opción PUTAUT especificada en el canal receptor o peticionario	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
DEF, 1 comprobación	-	CHL	CHL
DEF, 2 comprobaciones	-	CHL + MCA	CHL + MCA
CTX, 1 comprobación	CHL	CHL	CHL
CTX, 2 comprobaciones	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 comprobación	-	MCA	MCA
ONLYMCA, 2 comprobaciones	-	MCA	MCA
ALTMCA, 1 comprobación	MCA	MCA	MCA
ALTMCA, 2 comprobaciones	MCA	MCA	MCA + ALT

Clave:

MCA (ID de usuario de MCA)
El ID de usuario especificado para el atributo de canal MCAUSER en el receptor; si está en blanco, se utiliza el ID de usuario del espacio de direcciones del iniciador de canal del extremo receptor o peticionario.

CHL (ID de usuario de canal)
En TCP/IP, la seguridad no está soportada por el sistema de comunicación para el canal. Si se utiliza TLS (seguridad de la capa de transporte) y se ha transmitido un certificado digital desde el asociado, se utilizará el ID de usuario asociado a este certificado (si está instalado), o el ID de usuario asociado a un filtro coincidente encontrado mediante el Filtro de nombre de certificado (CNF) de RACF. Si no se encuentra ningún ID de usuario asociado, o si no se está utilizando TLS, el ID de usuario del espacio de direcciones del iniciador de canal del extremo receptor o peticionario se utiliza como ID de usuario de canal en los canales que están definidos con el parámetro PUTAUT establecido en DEF o CTX.

Nota: El uso del Filtro de nombre de certificado (CNF) de RACF le permite asignar el mismo ID de usuario RACF a varios usuarios remotos, por ejemplo a todos los usuarios de la misma unidad de organización, quienes, naturalmente, tendrían todos la misma autorización de seguridad. Esto significa que el servidor no necesita tener una copia del certificado de cada posible usuario remoto por todo el mundo, lo que simplifica enormemente la gestión y distribución de certificados.

Si el parámetro PUTAUT está establecido en ONLYMCA o ALTMCA para el canal, se hace caso omiso del ID de usuario de canal y se utiliza el ID de usuario de MCA del receptor o peticionario. Esto también se aplica a los canales TCP/IP que utilizan TLS.

ALT (ID de usuario alternativo)
El ID de usuario de la información de contexto (es decir, el campo *UserIdentifier*) dentro del descriptor de mensaje del mensaje. Este ID de usuario se mueve al campo *AlternateUserID* en el descriptor de objeto antes de que se emita una llamada **MQOPEN** o **MQPUT1** para la cola de destino.

z/OS *Canales receptores que utilizan LU 6.2*

Los ID de usuario que se comprueban dependen de la opción PUTAUT del canal y de si se va a realizar una o dos comprobaciones.

Tabla 62. Los ID de usuario que se comprueban con el nombre de perfil para canales LU 6.2

Opción PUTAUT especificada en el canal receptor o peticionario	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
DEF, 1 comprobación	-	CHL	CHL
DEF, 2 comprobaciones	-	CHL + MCA	CHL + MCA
CTX, 1 comprobación	CHL	CHL	CHL
CTX, 2 comprobaciones	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 comprobación	-	MCA	MCA
ONLYMCA, 2 comprobaciones	-	MCA	MCA
ALTMCA, 1 comprobación	MCA	MCA	MCA
ALTMCA, 2 comprobaciones	MCA	MCA	MCA + ALT

Clave:

MCA (ID de usuario de MCA)
El ID de usuario especificado para el atributo de canal MCAUSER en el receptor; si está en blanco, se utiliza el ID de usuario del espacio de direcciones del iniciador de canal del extremo receptor o peticionario.

CHL (ID de usuario de canal)

Canales peticionario-servidor
Si el canal se inicia desde el peticionario, no hay ninguna oportunidad de recibir un ID de usuario de red (el ID de usuario de canal).

Si el parámetro PUTAUT está establecido en DEF o CTX en el canal peticionario, el ID de usuario de canal es el del espacio de direcciones del iniciador de canal del peticionario porque no se recibe ningún ID de usuario de la red.

Si el parámetro PUTAUT está establecido en ONLYMCA o ALTMCA, se hace caso omiso del ID de usuario de canal y se utiliza el ID de usuario de MCA del peticionario.

Otros tipos de canal
Si el parámetro PUTAUT está establecido en DEF o CTX en el canal receptor o peticionario, el ID de usuario de canal es el ID de usuario que se recibe del sistema de comunicaciones cuando se inicia el canal.

- Si el canal emisor está en z/OS, el ID de usuario de canal que se recibe es el ID de usuario del espacio de direcciones del iniciador de canal del emisor.

- Si el canal emisor está en una plataforma diferente (por ejemplo, platform (for example, AIX), el ID de usuario de canal que se recibe lo proporciona normalmente el parámetro USERID de la definición de canal.

Si el ID de usuario que se recibe está en blanco, o no se recibe ningún ID de usuario, se utiliza un ID de usuario de canal de espacios en blanco.

ALT (ID de usuario alternativo)
El ID de usuario de la información de contexto (es decir, el campo *UserIdentifier*) dentro del descriptor de mensaje del mensaje. Este ID de usuario se mueve al campo *AlternateUserID* en el descriptor de objeto antes de que se emita una llamada MQOPEN o MQPUT1 para la cola de destino.

z/OS *Peticiones MQI de cliente*

Se pueden utilizar varios ID de usuario, dependiendo de los ID de usuario y las variables de entorno que se hayan establecido. Estos ID de usuario se comprueban con varios perfiles, dependiendo de la opción PUTAUT utilizada y de si se especifica un ID de usuario alternativo.

En esta sección se describen los ID de usuario que se comprueban para las peticiones MQI de cliente emitidas a través de canales de conexión con el servidor para TCP/IP y LU 6.2. El ID de usuario de MCA y el ID de usuario de canal son como los de los canales TCP/IP y LU 6.2 descritos en las secciones anteriores.

Para los canales de conexión con el servidor, se utiliza el ID de usuario recibido del cliente, si el atributo MCAUSER está en blanco.

Consulte “Control de accesos para clientes” en la página 98 para obtener más información.

Para peticiones **MQOPEN**, **MQSUB** y **MQPUT1** de cliente, utilice las reglas siguientes para determinar el perfil que se comprueba:

- Si la solicitud especifica autorización de usuario alternativo, se realiza una comprobación en *hlq*.ALTERNATE.USER. Perfil *userid* .
- Si la solicitud especifica autorización de contexto, se realiza una comprobación en *hlq*.CONTEXTO. Perfil *queuename* .
- Para todas las peticiones **MQOPEN**, **MQSUB** y **MQPUT1**, se realiza una comprobación en el perfil *hlq.resourcename*.

Cuando haya determinado qué perfiles se comprueban, utilice la tabla siguiente para determinar qué identificadores (ID) de usuario se comprueban con estos perfiles.

Tabla 63. Los ID de usuario que se comprueban con el nombre de perfil para canales de conexión con el servidor LU 6.2 y TCP/IP

Opción PUTAUT especificada en el canal de conexión con el servidor	¿Se ha especificado un ID de usuario alternativo en la apertura?	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
DEF, 1 comprobación	No	-	CHL	CHL
DEF, 1 comprobación	Sí	CHL	CHL	CHL

Tabla 63. Los ID de usuario que se comprueban con el nombre de perfil para canales de conexión con el servidor LU 6.2 y TCP/IP (continuación)

Opción PUTAUT especificada en el canal de conexión con el servidor	¿Se ha especificado un ID de usuario alternativo en la apertura?	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
DEF, 2 comprobaciones	No	-	CHL + MCA	CHL + MCA
DEF, 2 comprobaciones	Sí	CHL + MCA	CHL + MCA	CHL + ALT
ONLYMCA, 1 comprobación	No	-	MCA	MCA
ONLYMCA, 1 comprobación	Sí	MCA	MCA	MCA
ONLYMCA, 2 comprobaciones	No	-	MCA	MCA
ONLYMCA, 2 comprobaciones	Sí	MCA	MCA	MCA + ALT

Clave:

MCA (ID de usuario de MCA)
ID de usuario especificado para el atributo de canal MCAUSER en la conexión con servidor; si está en blanco, se utiliza el ID de usuario del espacio de direcciones del iniciador de canal.

CHL (ID de usuario de canal)
En TCP/IP, la seguridad no está soportada por el sistema de comunicación para el canal. Si se utiliza TLS (seguridad de la capa de transporte) y se ha transmitido un certificado digital desde el asociado, se utilizará el ID de usuario asociado a este certificado (si está instalado), o el ID de usuario asociado a un filtro coincidente encontrado mediante el Filtro de nombre de certificado (CNF) de RACF. Si no se encuentra ningún ID de usuario asociado, o si no se está utilizando TLS, el ID de usuario del espacio de direcciones del iniciador de canal se utiliza como ID de usuario de canal en los canales que están definidos con el parámetro PUTAUT establecido en DEF o CTX.

Nota: El uso del Filtro de nombre de certificado (CNF) de RACF le permite asignar el mismo ID de usuario RACF a varios usuarios remotos, por ejemplo a todos los usuarios de la misma unidad de organización, quienes, naturalmente, tendrían todos la misma autorización de seguridad. Esto significa que el servidor no necesita tener una copia del certificado de cada posible usuario remoto por todo el mundo, lo que simplifica enormemente la gestión y distribución de certificados.

Si el parámetro PUTAUT está establecido en ONLYMCA o ALTMCA para el canal, se hace caso omiso del ID de usuario de canal y se utiliza el ID de usuario de MCA del canal de conexión con el servidor. Esto también se aplica a los canales TCP/IP que utilizan TLS.

ALT (ID de usuario alternativo)
El ID de usuario de la información de contexto (es decir, el campo *UserIdentifier*) dentro del descriptor de mensaje del mensaje. Este ID de usuario se mueve al campo *AlternateUserID* en el descriptor de objeto o suscripción antes de que se emita una llamada **MQOPEN**, **MQSUB** o **MQPUT1** en nombre de la aplicación cliente.

z/OS *Ejemplo de iniciador de canal*
Un ejemplo de cómo se comprueban los ID de usuario con los perfiles RACF.

Un usuario realiza una operación **MQPUT1** para una cola del gestor de colas QM01 que se resuelve en una cola llamada QB del gestor de colas QM02. El mensaje se envía en un canal TCP/IP llamado QM01.TO.QM02. RESLEVEL está establecido en NONE, y la apertura se realiza con comprobación de contexto y de ID de usuario alternativo. La definición de canal receptor tiene PUTAUT(CTX) y el ID de usuario de MCA está establecido. ¿Qué identificadores (ID) de usuario se utilizan en el canal receptor para transferir el mensaje a la cola QB?

Respuesta: La Tabla 55 en la página 238 muestra que se comprueban dos ID de usuario porque RESLEVEL está establecido en NONE.

La Tabla 61 en la página 244 muestra que, con PUTAUT establecido en CTX y 2 comprobaciones, se comprueban los siguientes ID de usuario:

- El ID de usuario del iniciador de canal y el ID de usuario de MCAUSER se comprueban con el perfil hlq.ALTERNATE.USER.userid.
- El ID de usuario del iniciador de canal y el ID de usuario de MCAUSER se comprueban con el perfil hlq.CONTEXT.queuename.
- El ID de usuario del iniciador de canal y el ID de usuario alternativo especificado en el descriptor de mensaje (MQMD) se comprueban con el perfil hlq.Q2.

z/OS *Identificadores de usuario utilizados por el agente de transferencia a colas entre grupos*
Los ID de usuario que se comprueban cuando el agente de transferencia a colas entre grupos abre las colas de destino están determinados por los valores de los atributos del gestor de colas IGQAUT y IGQUSER.

Los ID de usuario posibles son:

ID de usuario de transferencia a colas entre grupos (IGQ)
El ID de usuario determinado por el atributo IGQUSER del gestor de colas receptor. Si éste está establecido en espacios en blanco, se utiliza el ID de usuario del gestor de colas receptor. Sin embargo, puesto que el gestor de colas receptor tiene autorización para acceder a todas las colas definidas en él, no se realizan comprobaciones de seguridad para el ID de usuario del gestor de colas receptor. En este caso:

- Si sólo se va a comprobar un ID de usuario y el ID de usuario es el del gestor de colas receptor, no se lleva a cabo ninguna comprobación de seguridad. Esto puede ocurrir cuando IGQAUT se establece en ONLYIGQ o ALTIGQ.
- Si se van a comprobar dos ID de usuario y uno de los ID de usuario es el del gestor de colas receptor, sólo se llevan a cabo comprobaciones de seguridad para el otro ID de usuario. Esto puede ocurrir cuando IGQAUT se establece en DEF, CTX o ALTIGQ.
- Si se van a comprobar dos ID de usuario y los dos ID de usuario son el del gestor de colas receptor, no se lleva a cabo ninguna comprobación de seguridad. Esto puede ocurrir cuando IGQAUT se establece en ONLYIGQ.

ID de usuario del gestor de colas emisor (SND)
El ID de usuario del gestor de colas dentro del grupo de compartición de colas que transfirió el mensaje a la cola SYSTEM.QSG.TRANSMIT.QUEUE.

ID de usuario alternativo (ALT)
El ID de usuario especificado en el campo *IdentificadorUsuario* del descriptor de mensaje del mensaje.

Tabla 64. Los ID de usuario que se comprueban con el nombre de perfil para la transferencia a colas entre grupos

Opción IGQAUT especificada en el gestor de colas receptor	Perfil hlq.ALTERNATE.USER.userid	Perfil hlq.CONTEXT.queuename	Perfil hlq.resourcename
DEF, 1 comprobación	-	SND	SND
DEF, 2 comprobaciones	-	SND +IGQ	SND +IGQ
CTX, 1 comprobación	SND	SND	SND
CTX, 2 comprobaciones	SND + IGQ	SND +IGQ	SND + ALT
ONLYIGQ, 1 comprobación	-	IGQ	IGQ
ONLYIGQ, 2 comprobaciones	-	IGQ	IGQ
ALTIGQ, 1 comprobación	-	IGQ	IGQ
ALTIGQ, 2 comprobaciones	IGQ	IGQ	IGQ + ALT

Clave:

ALT
ID de usuario alternativo.

IGQ
ID de usuario de IGQ.

SND
ID de usuario del gestor de colas emisor.

z/OS *Identificadores de usuario en blanco y niveles de UACC*

Si se especifica un ID de usuario en blanco, la sesión se inicia con un usuario no definido de RACF. No otorgue acceso de gran alcance al usuario no definido.

Pueden existir ID de usuario en blanco cuando un usuario está manipulando mensajes utilizando seguridad de contexto o de usuario alternativo, o cuando se pasa a IBM MQ un ID de usuario en blanco. Por ejemplo, se utiliza un ID de usuario en blanco cuando se graba un mensaje en la cola de entrada de mandatos del sistema sin contexto.

Nota: Un ID de usuario de "* " (es decir, un carácter asterisco seguido de siete espacios en blanco) se trata como un ID de usuario no definido.

IBM MQ pasa el ID de usuario en blanco a RACF y la sesión se inicia con un usuario no definido de RACF. Todas las comprobaciones de seguridad utilizan entonces el acceso universal (UACC) para el perfil correspondiente. Dependiendo de cómo haya establecido sus niveles de acceso, el UACC podría otorgar al usuario no definido un acceso de gran alcance.

Por ejemplo, si emite este mandato RACF desde TSO:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.EVERYONE UACC(UPDATE)
```

define un perfil que permite tanto a los ID de usuario definidos por z/OS(que no se han puesto en la lista de acceso) como al ID de usuario no definido de RACF transferir mensajes y obtener mensajes de esa cola.

Para protegerse frente a los ID de usuario en blanco, debe planificar cuidadosamente sus niveles de acceso, y limitar el número de personas que pueden utilizar la seguridad de contexto y de usuario alternativo. Debe impedir que las personas que utilicen el ID de usuario no definido de RACF obtengan acceso a recursos a los que no deben acceder. Sin embargo, al mismo tiempo, debe permitir el acceso a las personas con identificadores (ID) de usuario definidos. Para ello, puede especificar un ID de usuario de asterisco (*) en un mandato RACF PERMIT, otorgando acceso a recursos para todos los ID de usuario definidos. Por lo tanto, se deniega el acceso a todos los ID de usuario no definidos (como por ejemplo "* "). Por ejemplo, estos mandatos RACF impiden que el ID de usuario no definido de RACF obtenga acceso a la cola para transferir u obtener mensajes:

```
RDEFINE MQQUEUE Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY UACC(NONE)
PERMIT Q.AVAILABLE.TO.RACF.DEFINED.USERS.ONLY CLASS(MQQUEUE) ACCESS(UPDATE) ID(*)
```

z/OS Identificadores de usuarios de z/OS y autenticación de multifactores (MFA)

La autenticación de multifactores de IBM para z/OS permite a los administradores de sistemas de z/OS mejorar la autenticación SAF exigiendo a los usuarios identificados usar múltiples factores de autenticación (por ejemplo, una contraseña y un token criptográfico) para iniciar sesión en un sistema z/OS. La MFA de IBM también soporta tecnologías de generación de contraseñas únicas basadas en tiempo como, por ejemplo, SecureId de RSA.

La mayor parte del tiempo, IBM MQ no es consciente de cómo los usuarios han "iniciado sesión" en CICS o en los sistemas por lotes que impulsan el trabajo de IBM MQ, la credencial del ID del usuario que ha iniciado sesión está asociada a la tarea o al espacio de direcciones de z/OS y IBM MQ usa esto para comprobar la autorización a los recursos. Los ID de usuario habilitados para MFA se pueden usar en la autorización de recursos de IBM MQ y en la autenticación mediante tíquets de paso usados en los puentes de CICS e IMS.

Importante: No obstante, proceden consideraciones especiales cuando se usan aplicaciones como, por ejemplo, IBM MQ Explorer, que pasan credenciales de ID de usuario y contraseña en una llamada de API MQCONNX con la opción*MQCSP_AUTH_USER_ID_AND_PWD*. IBM MQ no tiene ningún servicio para pasar una credencial adicional en esta solicitud de API.

Las limitaciones y los métodos alternativos potenciales se describen en el texto siguiente.

IBM MQ Explorer

IBM MQ Explorer no se puede usar para iniciar sesión en un sistema z/OS con un ID de usuario para el que MFA esté autorizado, porque no hay ningún servicio para pasar un segundo factor de autenticación desde IBM MQ Explorer a z/OS.

Además, hay dos mecanismos diferentes utilizados por IBM MQ Explorer para reutilizar una credencial de ID de usuario y contraseña, que necesitan una atención especial cuando están en vigor las contraseñas únicas:

1. IBM MQ Explorer tiene la capacidad para almacenar contraseñas en un formato ofuscado en la máquina local para iniciar sesión en un momento posterior. Esta capacidad tiene que inhabilitarse haciendo que el explorador solicite una contraseña cada vez que se realiza una conexión con el gestor de colas de z/OS.

   Para ello, utilice el siguiente procedimiento:

a. Seleccione **Gestores de colas**.

b. En la lista visualizada, seleccione el gestor de colas que necesite y pulse con el botón derecho en dicho gestor de colas.

c. Seleccione **Detalles de conexión** en la lista de menú que aparece.

d. Seleccione **Propiedades** en la lista de menú siguiente y seleccione la pestaña **ID de usuario**.

Asegúrese de seleccionar el botón de opción **solicitar contraseña**.

2. Diversas operaciones en IBM MQ Explorer como, por ejemplo, la exploración de mensajes en colas, la prueba de suscripciones, etc. inician una nueva hebra que se autentica con IBM MQ usando la credencial usada por primera vez al iniciar sesión. Puesto que la credencial de contraseña no se puede reutilizar, no se pueden utilizar estas operaciones.

Hay dos posibles soluciones temporales a nivel de configuración MFA para estos problemas:

- Usar la exclusión de identificadores de aplicación de MFA para excluir las tareas de IBM MQ del procesamiento MFA.

  Para ello, emita los mandatos siguientes:

  1. ```
     RDEFINE MFADEF MFABYPASS.USERID.chinuser
     ```

     donde *chinuser* es el ID de usuario de nivel de espacio de direcciones del iniciador de canal (asociado con el iniciador de canal a través de la clase STC)

  2. ```
     PERMIT MFABYPASS.USERID.chinuser CLASS MFADEF ACCESS(READ) ID(explorer user)
     ```

  Para obtener información adicional sobre este enfoque, consulte Puenteado de IBM MFA en aplicaciones.

- Usar el soporte fuera de banda, que se introdujo en IBM MFA 1.2. Con este enfoque, se hace una autenticación previa en el servidor web IBM AMF, y, además del ID de usuario y la contraseña, se especifica una autenticación adicional como se determine por política. El servidor IBM MFA genera una credencial de token de caché que luego se especifica en el diálogo de autenticación de IBM MQ Explorer. El administrador de seguridad puede permitir que esta credencial se reproduzca durante un período de tiempo razonable, de modo que se posibilite un uso normal de IBM MQ Explorer.

Para obtener más información sobre este enfoque, consulte Introducción a IBM AMF.

z/OS Gestión de la seguridad de IBM MQ for z/OS

IBM MQ utiliza una tabla de almacenamiento para contener la información relativa a cada usuario y las peticiones de acceso realizadas por cada usuario. Para gestionar esta tabla eficazmente y para reducir el número de peticiones realizadas desde IBM MQ al gestor de seguridad externo (ESM), hay disponible una serie de controles.

Estos controles están disponibles mediante los paneles de operaciones y paneles de control y los mandatos de IBM MQ.

z/OS *Reverificación del ID de usuario*

Si la definición de RACF de un usuario que está utilizando recursos de IBM MQ ha cambiado, por ejemplo conectando el usuario a un nuevo grupo, puede indicar al gestor de colas que vuelva a iniciar la sesión de este usuario la próxima vez que intente acceder a un recurso de IBM MQ. Puede hacerlo mediante el mandato de IBM MQ RVERIFY SECURITY.

- El usuario HX0804 está obteniendo y transfiriendo mensajes a las colas PAYROLL del gestor de colas PRD1. Sin embargo, HX0804 ahora necesita acceso a algunas de las colas PENSION del mismo gestor de colas (PRD1).
- El administrador de seguridad de datos conecta al usuario HX0804 al grupo RACF que permite el acceso a las colas PENSION.

- Para que HX0804 pueda acceder a las colas PENSION inmediatamente (es decir, sin cerrar el gestor de colas PRD1 o esperar a que HX0804 supere el tiempo de espera) debe utilizar el mandato de IBM MQ:

```
RVERIFY SECURITY(HX0804)
```

Nota: Si desactiva el tiempo de espera de ID de usuario durante largos periodos de tiempo (días o incluso semanas) mientras el gestor de colas está en ejecución, recuerde que debe ejecutar el mandato RVERIFY SECURITY para todos los usuarios que se hayan revocado o suprimido durante dicho tiempo.

z/OS *Tiempos de espera de ID de usuario*

Puede hacer que IBM MQ desconecte a un usuario de un gestor de colas tras un periodo de inactividad.

Cuando un usuario accede a un recurso de IBM MQ, el gestor de colas intenta conectar este usuario en el gestor de colas (si la seguridad de subsistema está activa). Esto significa que el usuario es autenticado en el ESM. Este usuario permanece conectado en IBM MQ hasta que el gestor de colas se concluye, o hasta que se *excede el tiempo de espera* del ID de usuario (la autenticación caduca) o se vuelve a verificar (reautenticar) el ID de usuario.

Cuando se excede el tiempo de espera de un usuario, el ID de usuario es *desconectado* en el gestor de colas y se descarta cualquier información relacionada con la seguridad conservada para este usuario. La conexión y desconexión del usuario dentro del gestor de colas no es evidente para el programa de aplicación o el usuario.

Los usuarios son susceptibles de exceder el tiempo de espera cuando no han utilizado ningún recurso de IBM MQ durante un periodo de tiempo predeterminado. Este periodo de tiempo se establece mediante el mandato MQSC ALTER SECURITY.

Se pueden especificar dos valores en el mandato ALTER SECURITY:

TIMEOUT
El periodo de tiempo, en minutos, que un ID de usuario no utilizado y sus recursos asociados pueden permanecen en el gestor de colas IBM MQ.

INTERVAL
El periodo de tiempo, en minutos, entre las comprobaciones de los ID de usuario y sus recursos asociados, para determinar si *TIMEOUT* ha caducado.

Por ejemplo, si el valor *TIMEOUT* es 30 y el valor *INTERVAL* es 10, cada 10 minutos IBM MQ comprueba los ID de usuario y sus recursos asociados para determinar si alguno no se ha utilizado durante 30 minutos. Si se encuentra un ID de usuario que haya excedido el tiempo de espera,se desconectará del gestor de colas. Si se encuentra información de recursos que ha excedido el tiempo de espera que está asociada a identificadores (ID) de usuario que no lo han excedido, esa información de recursos se descarta. Si no desea desconectar a los ID de usuario por tiempo de espera excedido, establezca el valor *INTERVAL* en cero. No obstante, si el valor *INTERVAL* es cero, el almacenamiento ocupado por los ID de usuario y sus recursos asociados no se libera hasta que se emite un mandato **REFRESH SECURITY** o **RVERIFY SECURITY**.

El ajuste de este valor puede ser importante si tiene muchos usuarios ocasionales. Si establece valores de intervalo y tiempo de espera pequeños, los recursos que ya no se necesitan se liberan.

Nota: Si utiliza valores para *INTERVAL* o *TIMEOUT* que no sean los predeterminados, debe volver a emitir el mandato cada vez que inicie el gestor de colas. Puede hacer esto automáticamente colocando el mandato **ALTER SECURITY** en el conjunto de datos CSQINP1 para dicho gestor de colas.

z/OS *Renovación de la seguridad del gestor de colas en z/OS*

IBM MQ for z/OS almacena en la memoria caché los datos de RACF para mejorar el rendimiento. Cuando cambie ciertas clases de seguridad, debe renovar esta información almacenada en memoria caché. Renueve la seguridad ocasionalmente, por razones de rendimiento. También puede decidir renovar sólo la información de seguridad TLS.

Cuando se abre una cola por primera vez (o por primera vez desde una renovación de seguridad), IBM MQ realiza una comprobación RACF para obtener los derechos de acceso del usuario y coloca esta

información en la memoria caché. Los datos almacenados en la memoria caché son los ID de usuario y los recursos en los que se ha realizado la comprobación de seguridad. Si la cola la abre de nuevo el mismo usuario, la presencia de los datos almacenados en memoria caché significa que IBM MQ no tiene que emitir comprobaciones RACF, lo cual mejora el rendimiento. La acción de una renovación de seguridad es descartar cualquier información de seguridad almacenada en la memoria caché y obligar así a IBM MQ a realizar una nueva comprobación en RACF. Siempre que añada, cambie o suprima un perfil de recurso RACF que esté contenido en la clase MQADMIN, MXADMIN, MQPROC, MXPROC, MQQUEUE, MXQUEUE, MQNLIST, MXNLIST o MXTOPIC, debe indicar a los gestores de colas que utilizan esta clase que renueven la información de seguridad que contienen. Para ello, emita los mandatos siguientes:

- El mandato RACF SETROPTS RACLIST(nombreclase) REFRESH para renovar a nivel de RACF.
- El mandato IBM MQ REFRESH SECURITY para renovar la información de seguridad mantenida por el gestor de colas. Este mandato debe ser emitido por cada gestor de colas que acceda a los perfiles que han cambiado. Si tiene un grupo de compartición de colas, puede utilizar el atributo de ámbito de mandato para dirigir el mandato a todos los gestores de colas del grupo.

  Nota: Si ha conectado un nuevo usuario a un grupo existente, debe ejecutar el mandato IBM MQ RVERIFY SECURITY(ID de usuario). El mandato REFRESH SECURITY (*) no permite que el gestor de colas vuelva a firmar este usuario, la próxima vez que intente acceder a un recurso IBM MQ .

Si está utilizando perfiles genéricos en cualquiera de las clases IBM MQ, también debe emitir mandatos de renovación RACF normales, si cambia, añade o suprime cualquier perfil genérico. Por ejemplo, SETROPTS GENERIC(nombreclase) REFRESH.

Sin embargo si se añade, cambia o suprime un perfil de recurso de RACF y todavía no se ha accedido al recurso al que se aplica (por lo que no hay información almacenada en memoria caché), IBM MQ utiliza la nueva información de RACF sin que se emita un mandato REFRESH SECURITY.

Si la auditoría de RACF está activada (por ejemplo, mediante el mandato de RACF RALTER AUDIT(intento-acceso (nivel_acceso_auditoría)), no se lleva a cabo el almacenamiento en memoria caché y, por lo tanto, IBM MQ consulta directamente el espacio de datos de RACF para cada comprobación. Los cambios, por tanto, se recuperan inmediatamente y no es necesario el mandato REFRESH SECURITY para acceder a los cambios. Puede comprobar si la auditoría RACF está activada mediante el mandato RACF RLIST. Por ejemplo, podría emitir el mandato

```
RLIST MQQUEUE (qmgr.SYSTEM.COMMAND.INPUT) GEN
```

y recibir los resultados

```
CLASS      NAME
-----      ----
MQQUEUE    QP*.SYSTEM.COMMAND.*.** (G)
    AUDITING
    --------
    FAILURES(READ)
```

Esto indica que la auditoría está activada. Para obtener más información, consulte las publicaciones *z/OS Security Server RACF Auditor's Guide* y *z/OS Security Server RACF Command Language Reference*.

La Figura 17 en la página 254 resume las situaciones en que se almacena información de seguridad en la memoria caché y en las que se utiliza la información almacenada en memoria caché.

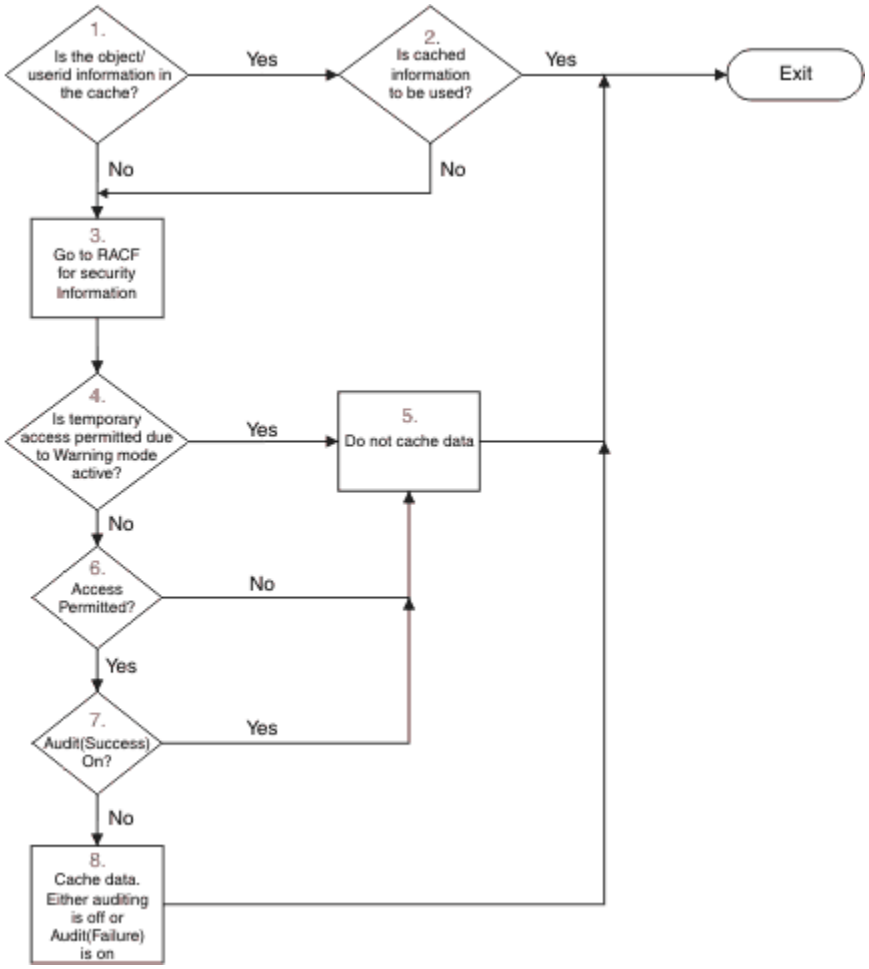


Figura 17. Flujo lógico para el almacenamiento en memoria caché de seguridad IBM MQ

Si cambia los valores de seguridad mediante la adición o supresión de perfiles de conmutador en las clases MQADMIN o MXADMIN, utilice uno de estos mandatos para recuperar estos cambios dinámicamente:

```
REFRESH SECURITY(*)
REFRESH SECURITY(MQADMIN)
REFRESH SECURITY(MXADMIN)
```

Esto significa que puede activar nuevos tipos de seguridad, o desactivarlos sin tener que reiniciar el gestor de colas.

Por razones de rendimiento, estas son las únicas clases afectadas por el mandato REFRESH SECURITY. No necesita utilizar REFRESH SECURITY si cambia un perfil en las clases MQCONN o MQCMDS.

Nota: No es necesaria una renovación de la clase MQADMIN o MXADMIN si cambia un perfil de seguridad RESLEVEL.

Por razones de rendimiento, utilice REFRESH SECURITY con la menor frecuencia posible, preferiblemente en horas de menor actividad. Puede minimizar el número de renovaciones de seguridad si conecta usuarios a grupos RACF que ya estén en la lista de acceso para perfiles IBM MQ en lugar de poner usuarios individuales en las listas de acceso. De este modo, se cambia el usuario en lugar del perfil de recurso. También puede emitir el mandato RVERIFY SECURITY para el usuario adecuado, en lugar de renovar la seguridad.

Como ejemplo de REFRESH SECURITY, suponga que define los nuevos perfiles para proteger el acceso a las colas que empiezan por INSURANCE.LIFE en el gestor de colas PRMQ. Utilice estos mandatos RACF:

```
RDEFINE MQQUEUE PRMQ.INSURANCE.LIFE.** UACC(NONE)
PERMIT PRMQ.INSURANCE.LIFE.** ID(LIFEGRP) ACCESS(UPDATE)
```

Debe emitir el siguiente mandato para indicar a RACF que renueve la información de seguridad que contiene, por ejemplo:

```
SETROPTS RACLIST(MQQUEUE) REFRESH
```

Puesto que estos perfiles son genéricos, debe indicar a RACF que renueve los perfiles genéricos para MQQUEUE. Por ejemplo:

```
SETROPTS GENERIC(MQQUEUE) REFRESH
```

A continuación, debe utilizar este mandato para informar al gestor de colas PRMQ de que los perfiles de cola han cambiado:

```
REFRESH SECURITY(MQQUEUE)
```

Renovación de la seguridad SSL/TLS

Para renovar la vista almacenada en memoria caché del Repositorio de claves TLS, emita el mandato REFRESH SECURITY con la opción TYPE(SSL). Esto le permite actualizar algunos de los valores TLS sin tener que reiniciar el iniciador de canal.

z/OS *Visualización del estado de la seguridad*

Para visualizar el estado de los conmutadores de seguridad, y de otros controles de seguridad, emita el mandato MQSC DISPLAY SECURITY.

La figura siguiente muestra la salida típica del mandato DISPLAY SECURITY ALL.

```
CSQH015I +CSQ1 Security timeout = 54 MINUTES
CSQH016I +CSQ1 Security interval = 12 MINUTES
CSQH030I +CSQ1 Security switches ...
CSQH034I +CSQ1 SUBSYSTEM: ON, 'SQ05.NO.SUBSYS.SECURITY' not found
CSQH032I +CSQ1 QMGR: ON, 'CSQ1.YES.QMGR.CHECKS' found
CSQH031I +CSQ1 QSG: OFF, 'SQ05.NO.QSG.CHECKS' found
CSQH031I +CSQ1 CONNECTION: OFF, 'CSQ1.NO.CONNECT.CHECKS' found
CSQH034I +CSQ1 COMMAND: ON, 'CSQ1.NO.COMMAND.CHECKS' not found
CSQH031I +CSQ1 CONTEXT: OFF, 'CSQ1.NO.CONTEXT.CHECKS' found
CSQH034I +CSQ1 ALTERNATE USER: ON, 'CSQ1.NO.ALTERNATE.USER.CHECKS' not found
CSQH034I +CSQ1 PROCESS: ON, 'CSQ1.NO.PROCESS.CHECKS' not found
CSQH034I +CSQ1 NAMELIST: ON, 'CSQ1.NO.NLIST.CHECKS' not found
CSQH034I +CSQ1 QUEUE: ON, 'CSQ1.NO.QUEUE.CHECKS' not found
CSQH034I +CSQ1 TOPIC: ON, 'CSQ1.NO.TOPIC.CHECKS' not found
CSQH031I +CSQ1 COMMAND RESOURCES: OFF, 'CSQ1.NO.CMD.RESC.CHECKS' found
CSQ9022I +CSQ1 CSQHPDTC ' DISPLAY SECURITY' NORMAL COMPLETION
```

Figura 18. Salida típica del mandato DISPLAY SECURITY

El ejemplo muestra que el gestor de colas que respondió al mandato tiene activa la seguridad de subsistema, de mandatos, de usuario alternativo, de procesos, de listas de nombres y de colas a nivel de gestor de colas, pero no a nivel de grupo de compartición de colas. La seguridad de conexión, de recursos de mandatos y de contexto no están activas. También muestra que los tiempos de espera de ID de usuario están activos, y que cada 12 minutos el gestor de colas comprueba si hay identificadores (ID) de usuario que no se han utilizado en este gestor de colas durante 54 minutos y los elimina.

Nota: Este mandato muestra el estado actual de la seguridad. No refleja necesariamente el estado actual de los perfiles de conmutador definidos para RACF ni el estado de las clases RACF. Por ejemplo, es posible que se hayan cambiado los perfiles de conmutador desde el último reinicio de este gestor de colas o mandato REFRESH SECURITY.

z/OS Tareas de instalación de seguridad para z/OS

Después de instalar y personalizar IBM MQ, otorgue autorización a los procedimientos de tarea iniciada en RACF, autorice el acceso a diversos recursos y establezca definiciones RACF. Opcionalmente, configure el sistema para TLS.

Cuando IBM MQ se instala y personaliza por primera vez, debe realizar estas tareas relacionadas con la seguridad:

1. Configurar la seguridad del sistema y de los conjuntos de datos de IBM MQ mediante las siguientes acciones:
   - Autorizar al procedimiento de tarea iniciada de gestor de colas xxxxMSTR y al procedimiento de tarea iniciada de gestión de colas distribuidas xxxxCHIN a que se ejecuten bajo RACF.
   - Autorizar el acceso a los conjuntos de datos del gestor de colas.
   - Autorizar el acceso a los recursos para los ID de usuario que utilizarán el gestor de colas y los programas de utilidad.
   - Autorizar el acceso para los gestores de colas que utilizarán las estructuras de lista de recurso de acoplamiento.
   - Autorizar el acceso para los gestores de colas que utilizarán Db2.
2. Establecer definiciones RACF para la seguridad de IBM MQ.
3. Si desea utilizar TLS (seguridad de la capa de transporte), prepare el sistema para utilizar certificados y claves.

z/OS *Configuración de la seguridad de los conjuntos de datos de IBM MQ for z/OS*

Hay muchos tipos de usuarios de IBM MQ. Utilice RACF para controlar su acceso a los conjuntos de datos del sistema.

Entre los posibles usuarios de los conjuntos de datos de IBM MQ se incluyen las siguientes entidades:

- El propio gestor de colas.
- El iniciador de canal
- Administradores de IBM MQ que necesiten crear conjuntos de datos de IBM MQ, ejecutar programas de utilidad y tareas similares.
- Programadores de aplicaciones que necesiten utilizar los libros de copias proporcionados por IBM MQ, incluir conjuntos de datos, macros y recursos similares.
- Aplicaciones en las que participen uno o varios:
  - Trabajos por lotes
  - Usuarios TSO
  - Regiones de CICS
  - Regiones de IMS
- Conjuntos de datos CSQOUTX y CSQSNAP
- Colas dinámicas SYSTEM.CSQXCMD.*

Para todos estos posibles usuarios, proteja los conjuntos de datos de IBM MQ con RACF.

También debe controlar el acceso a todos los conjuntos de datos 'CSQINP'.

z/OS *Autorización RACF de procedimientos de tarea iniciada*

Algunos conjuntos de datos de IBM MQ son para uso exclusivo del gestor de colas. Si protege los conjuntos de datos de IBM MQ utilizando RACF, también debe autorizar el procedimiento de tarea iniciada de gestor de colas xxxxMSTR, y el procedimiento de tarea iniciada de gestión de colas distribuidas xxxxCHIN, utilizando RACF. Para hacer esto, utilice la clase STARTED. De forma alternativa, puede utilizar la tabla de procedimientos iniciados (ICHRIN03), pero entonces deberá hacer una IPL del sistema z/OS para que los cambios tengan efecto.

Para obtener más información, consulte la publicación *z/OS Security Server RACF System Programmer's Guide*.

El ID de usuario RACF identificado debe tener el acceso necesario a los conjuntos de datos en el procedimiento de tarea iniciada. Por ejemplo, si asocia un procedimiento de tarea iniciada de gestor de colas llamado CSQ1MSTR al ID de usuario de RACF QMGRCSQ1, el ID de usuario QMGRCSQ1 debe tener acceso a los recursos z/OS a los que accede el gestor de colas CSQ1.

Además, el contenido del campo GROUP en el ID de usuario del gestor de colas debe ser el mismo que el contenido del campo GROUP en el perfil STARTED para dicho gestor de colas. Si el contenido de cada campo GROUP no coincide, se impide al ID de usuario pertinente entrar en el sistema. Esta situación hace que IBM MQ se ejecute con un ID de usuario no definido y, por consiguiente, se cierre debido a una violación de seguridad.

Los ID de usuario RACF asociados a los procedimientos de tarea iniciada del gestor de colas y el iniciador de canal no debe tener establecido el atributo TRUSTED.

z/OS *Autorizar el acceso a los conjuntos de datos*

Los conjuntos de datos de IBM MQ deberían protegerse para que ningún usuario que no esté autorizado pueda ejecutar una instancia de gestor de colas o acceder a los datos del gestor de colas. Para ello, utilice la protección de conjunto de datos normal de z/OS RACF.

La Tabla 65 en la página 257 resume el acceso de RACF que el procedimiento de tarea iniciada del gestor de colas debe tener a los diferentes conjuntos de datos.

Tabla 65. Acceso de RACF a los conjuntos de datos asociados a un gestor de colas

acceso a RACF	**Conjuntos de datos**
READ	• `thlqual.SCSQAUTH` y `thlqual.SCSQANLx` (donde x es la letra de idioma para su idioma nacional). • Los conjuntos de datos a los que hacen referencia CSQINP1, CSQINP2 y CSQXLIB en el procedimiento de tarea iniciada del gestor de colas • Conjuntos de datos SMDS propiedad de otros gestores de colas del grupo. • Conjuntos de datos de registro, BSDS y de registro de archivado para otros gestores de colas del grupo.
UPDATE	• Todos los conjuntos de páginas y conjuntos de datos de anotaciones y BSDS. • Conjuntos de datos SMDS propiedad de un gestor de colas
ALTER	• Todos los conjuntos de datos de archivado.

La Tabla 66 en la página 258 resumen el acceso de RACF que el procedimiento de tarea iniciada para gestión de colas distribuidas debe tener a los diferentes conjuntos de datos.

Tabla 66. Acceso de RACF a los conjuntos de datos asociados a la gestión de colas distribuidas	
acceso a RACF	**Conjuntos de datos**
READ	• thlqual.SCSQAUTH, thlqual.SCSQANLx (donde x es la letra de idioma de su idioma nacional) y thlqual.SCSQMVR1. • Conjuntos de datos de la biblioteca LE. • Los conjuntos de datos a los que hacen referencia CSQXLIB y CSQINPX en el procedimiento de tarea iniciada del iniciador de canal.
UPDATE	• Conjuntos de datos CSQOUTX y CSQSNAP

Para obtener más información, consulte la publicación *z/OS Security Server RACF Security Administrator's Guide*.

z/OS V 9.1.4 *Cifrado de conjunto de datos*

Los conjuntos de datos de IBM MQ se pueden cifrar con el cifrado de conjunto de datos de z/OS, de modo que los datos estén protegidos, o por motivos normativos.

Puede proteger todos los conjuntos de páginas, el registro activo, el registro de archivado y los conjuntos de datos de rutina de carga (BSDS) con el cifrado de conjunto de datos de z/OS.

Atención: No puede proteger los conjuntos de datos de mensajes compartidos (SMDS) con el cifrado de conjuntos de datos de z/OS mediante IBM MQ for z/OS 9.1.3 o anterior.

Consulte la sección Confidencialidad para los datos en reposo en IBM MQ for z/OS con cifrado de conjunto de datos. para obtener más información.

z/OS *Configuración de la seguridad de recursos de IBM MQ for z/OS*

Hay muchos tipos de usuarios de IBM MQ. Utilice RACF para controlar su acceso a los recursos de IBM MQ.

Entre los posibles usuarios de los recursos de IBM MQ, tales como colas y canales, se incluyen las siguientes entidades:

- El propio gestor de colas.
- El iniciador de canal
- Administradores de IBM MQ que necesiten crear conjuntos de datos de IBM MQ, ejecutar programas de utilidad y tareas similares
- Programadores de aplicaciones que necesiten utilizar los libros de copias proporcionados por IBM MQ, incluir conjuntos de datos, macros y recursos similares.
- Aplicaciones en las que participen uno o varios:
  - Trabajos por lotes
  - Usuarios TSO
  - Regiones de CICS
  - Regiones de IMS
- Conjuntos de datos CSQOUTX y CSQSNAP
- Colas dinámicas SYSTEM.CSQXCMD.*

Para todos estos posibles usuarios, proteja los recursos de IBM MQ con RACF. En particular, tenga en cuenta que el iniciador de canal necesita tener acceso a diversos recursos, como se describe en las “Consideraciones de seguridad para el iniciador de canal en z/OS” en la página 265, por lo que el ID de usuario bajo el que se ejecuta debe estar autorizado para acceder a estos recursos.

Si está utilizando un grupo de compartición de colas, el gestor de colas puede emitir varios mandatos internamente, por lo que el ID de usuario que éste utiliza debe estar autorizado para emitir dichos mandatos. Los mandatos son:

- DEFINE, ALTER y DELETE para cada objeto que tiene QSGDISP(GROUP)
- START y STOP CHANNEL para cada canal utilizado con CHLDISP(SHARED)

z/OS *Configuración del sistema z/OS para utilizar TLS*

Utilice este tema como ejemplo de cómo configurar IBM MQ for z/OS con Transport Layer Security (TLS) utilizando mandatos RACF.

Si desea utilizar TLS para la seguridad de canal, hay una serie de tareas que debe realizar en el sistema. (Para más información sobre el uso de mandatos RACF para certificados y repositorios de claves (conjuntos de claves), consulte Trabajar con TLS en z/OS).

1. Cree un conjunto de claves en RACF para que contenga todas las claves y certificados para el sistema, mediante el mandato RACF RACDCERT. Por ejemplo:

   ```
   RACDCERT ID(CHINUSER) ADDRING(QM1RING)
   ```

   El ID debe ser el ID de usuario del espacio de direcciones del iniciador de canal o el ID de usuario que quiere que sea el propietario del conjunto de claves, si éste va a ser un conjunto de claves compartido.

2. Cree un certificado digital para cada gestor de colas, mediante el mandato RACF RACDCERT.

   La etiqueta del certificado debe ser el valor del atributo de IBM MQ **CERTLABL**, si éste está establecido, o el valor predeterminado `ibmWebSphereMQ` con el nombre del gestor de colas o el grupo de compartición de colas añadido. Para obtener información detallada, consulte Etiquetas de certificados digitales. En este ejemplo es `ibmWebSphereMQQM1`.

   Por ejemplo:

   ```
   RACDCERT ID(USERID) GENCERT
   SUBJECTSDN(CN('username') O('IBM') OU('departmentname') C('England'))
   WITHLABEL('ibmWebSphereMQQM1')
   ```

3. Conecte el certificado en RACF al conjunto de claves, mediante el mandato RACF RACDCERT. Por ejemplo:

   ```
   RACDCERT CONNECT(ID(USERID) LABEL('ibmWebSphereMQQM1') RING(QM1RING))
   CONNECT ID(CHINUSER)
   ```

   También tiene que conectar todos los certificados de firmante pertinentes (de una entidad emisora de certificados) al conjunto de claves. Es decir, todas las entidades emisoras de certificados para el certificado TLS de este gestor de colas y todas las entidades emisoras de certificados para todos los certificados TLS con las que se comunica este gestor de colas. Por ejemplo:

   ```
   RACDCERT ID(CHINUSER)
   CONNECT(CERTAUTH LABEL('My CA') RING(QM1RING) USAGE(CERTAUTH))
   ```

4. En cada uno de los gestores de colas, utilice el mandato de IBM MQ ALTER QMGR para especificar el repositorio de claves al que el gestor de colas debe apuntar. Por ejemplo, si el conjunto de claves es propiedad del espacio de direcciones del iniciador de canal:

```
ALTER QMGR SSLKEYR(QM1RING)
```

o si está utilizando un conjunto de claves compartido:

```
ALTER QMGR SSLKEYR(userid/QM1RING)
```

donde *idusuario* es el ID de usuario propietario del conjunto de claves compartido.

5. Las listas de revocación de certificados (CRL) permiten a entidades emisoras de certificados revocar los certificados que ya no son de confianza. Las CRL se almacenan en servidores LDAP. Para acceder a esta lista en el servidor LDAP, primero tiene que crear un objeto AUTHINFO con AUTHTYPE CRLLDAP, mediante el mandato de IBM MQ DEFINE AUTHINFO. Por ejemplo:

```
DEFINE AUTHINFO(LDAP1)
AUTHTYPE(CRLLDAP)
CONNAME(ldap.server(389))
LDAPUSER('')
LDAPPWD('')
```

En este ejemplo, la lista de revocación de certificados se almacena en un área pública del servidor LDAP, por lo que los campos LDAPUSER y LDAPPWD no son necesarios.

A continuación, coloque el objeto AUTHINFO en una lista de nombres, mediante el mandato de IBM MQ DEFINE NAMELIST. Por ejemplo:

```
DEFINE NAMELIST(LDAPNL) NAMES(LDAP1)
```

Por último, asocie la lista de nombres a cada gestor de colas, mediante el mandato de IBM MQ ALTER QMGR. Por ejemplo:

```
ALTER QMGR SSLCRLNL(LDAPNL)
```

6. Configure el gestor de colas para que ejecute llamadas TLS, mediante el mandato de IBM MQ ALTER QMGR. Esto define subtareas de servidor que sólo manejan llamadas SSL, lo que permite que los asignadores normales continúen procesándose de la forma habitual, sin que se vean afectados por las llamadas SSL. Debe tener al menos dos de esta subtareas. Por ejemplo:

```
ALTER QMGR SSLTASKS(8)
```

Este cambio sólo entra en vigor cuando se reinicia el iniciador de canal.

7. Establezca la especificación de cifrado que se utilizará para cada canal, con el mandato de IBM MQ DEFINE CHANNEL o ALTER CHANNEL. Por ejemplo:

```
ALTER CHANNEL(LDAPCHL)
CHLTYPE(SDR)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
```

Los dos extremos del canal deben tener establecida la misma especificación de cifrado.

z/OS Gestión de registros de autenticación de canal en un QSG

Los registros de autenticación de canal se aplican al gestor de colas en el que se han creado, no se comparten con todo el grupo de compartición de colas (QSG). Por consiguiente, si es preciso que todos los gestores de colas del grupo de compartición de colas tengan las mismas reglas, se tiene que realizar alguna tarea de gestión para que todas las reglas sean coherentes.

1. Añada siempre la opción `CMDSCOPE(*)` a todos los mandatos `SET CHLAUTH`. El mandato se enviará a todos los gestores de colas en ejecución del grupo de compartición de colas.
2. Utilice el mandato `DISPLAY CHLAUTH` con la opción `CMDSCOPE(*)` y a continuación, analice la respuesta para ver si los registros son los mismos en todos los gestores de colas. Cuando se encuentra una incoherencia, se puede emitir un mandato `SET CHLAUTH` que contenga la misma regla con `CMDSCOPE(*)` o `CMDSCOPE(qmgr-name)`.
3. Añada un miembro a la concatenación CSQINP2 del gestor de colas (para obtener detalles consulte Mandatos de inicialización) que tenga el conjunto completo de reglas. Estas se leerán como parte del proceso de inicialización del gestor de colas. Si el mandato `SET CHLAUTH` utiliza `ACTION(ADD)`, la regla sólo se añadirá si no existía. El uso de `ACTION(REPLACE)` sustituirá una regla existente si ya existe o la añadirá si no existe. A continuación, podría colocarse el mismo miembro en la concatenación CSQINP2 de todos los gestores de colas del grupo de compartición de colas.
4. Utilice el programa de utilidad CSQUTIL (para obtener detalles, consulte Emitir mandatos a IBM MQ (COMMAND)) para extraer las reglas de un gestor de colas utilizando la opción `MAKEDEF` o `MAKEREP`. A continuación, reproduzca la salida utilizando CSQUTIL en el gestor de colas de destino.

Conceptos relacionados

Registros de autenticación de canal

Para ejercer un control más preciso sobre el acceso otorgado a la conexión de sistemas a nivel de canal, puede utilizar registros de autenticación de canal.

z/OS Consideraciones de auditoría en z/OS

Los controles de auditoría RACF normales están disponibles para realizar una auditoría de seguridad de un gestor de colas. IBM MQ no recopila estadísticas de seguridad propias. Las únicas estadísticas son las que se pueden crear con la auditoría.

La auditoría RACF puede basarse en:

- ID de usuario
- Clases de recursos
- Perfiles

Para obtener más información, consulte la publicación *z/OS Security Server RACF Auditor's Guide*.

Nota: La auditoría degrada el rendimiento; cuanta más auditoría implemente, más rendimiento se degrada. Esta es también una consideración para el uso de la opción WARNING de RACF.

z/OS *Auditoría de RESLEVEL*

Utilice el parámetro del sistema RESAUDIT para controlar la producción de registros de auditoría RESLEVEL. Se producen registros de auditoría RACF GENERAL.

Para producir registros de auditoría RESLEVEL, establezca el parámetro del sistema RESAUDIT en YES. Si el parámetro RESAUDIT se establece en NO, no se producen registros de auditoría. Para obtener más información sobre el establecimiento de este parámetro, consulte Utilización de CSQ6SYSP.

Si RESAUDIT se establece en YES, no se toman registros de auditoría RACF normales cuando se realiza la comprobación RESLEVEL para ver qué acceso tiene un ID de usuario del espacio de direcciones al perfil hlq.RESLEVEL. En su lugar, IBM MQ solicita que RACF cree un registro de auditoría GENERAL (número

de suceso 27). Estas comprobaciones sólo se llevan a cabo en el momento de la conexión, por lo que el coste de rendimiento es mínimo.

Puede generar un informe de los registros de auditoría generales de IBM MQ utilizando el generador de informes de RACF (RACFRW). Puede utilizar los siguientes mandatos RACFRW para generar el informe del acceso RESLEVEL:

```
RACFRW
SELECT PROCESS
EVENT GENERAL
LIST
END
```

En Figura 19 en la página 262se muestra un informe de ejemplo de RACFRW, excluyendo los campos *Date*, *Time*y *SYSID* .

```
       RACF REPORT - LISTING OF PROCESS RECORDS                                  PAGE   4
                                E
                                V  Q
                                E  U
*JOB/USER *STEP/  --TERMINAL--  N  A
   NAME    GROUP     ID    LVL  T  L

 WS21B    MQMGRP IGJZM000   0   27 0  JOBID=(WS21B 05.111 09:44:57),USERDATA=()
    TRUSTED USER                      AUTH=(NONE),REASON=(NONE)
                                SESSION=TSOLOGON,TERMINAL=IGJZM000,
                                LOGSTR='CSQH RESLEVEL CHECK PERFORMED AGAINST
PROFILE(QM66.RESLEVEL),
                                CLASS(MQADMIN), ACCESS EQUATES TO
(CONTROL)',RESULT=SUCCESS,MQADMIN
```

Figura 19. Salida de ejemplo de RACFRW que muestra registros de auditoría generales RESLEVEL

A partir de la comprobación de los datos LOGSTR de esta salida de ejemplo, puede ver que el usuario TSO WS21B tiene acceso CONTROL a QM66.RESLEVEL. Esto significa que todas las comprobaciones de seguridad de recursos se pasan por alto cuando el usuario WS21B accede a los recursos de QM66.

Para obtener más información sobre cómo utilizar RACFRW, consulte *z/OS Security Server RACF Auditor's Guide*.

z/OS Personalización de la seguridad

Si desea cambiar la forma en que funciona la seguridad de IBM MQ, debe hacerlo mediante la salida SAF (ICHRFR00), o las salidas de su gestor de seguridad externo (ESM).

Para obtener más información sobre las salidas RACF, consulte el manual *z/OS Security Server RACROUTE Macro Reference*.

Nota: Debido a que IBM MQ optimiza las llamadas al ESM, es posible que no realicen peticiones RACROUTE en, por ejemplo, cada apertura de una determinada cola realizada por un determinado usuario.

z/OS Mensajes de violación de seguridad en z/OS

Una violación de seguridad se indica mediante el código de retorno MQRC_NOT_AUTHORIZED en un programa de aplicación, o mediante un mensaje en el registro de trabajo.

Puede devolverse un código de retorno MQRC_NOT_AUTHORIZED a un programa de aplicación por las siguientes razones:

- Un usuario no tiene autorización para conectarse al gestor de colas. En este caso, recibe un mensaje ICH408I en el registro de trabajo Batch/TSO, CICS o IMS.

- Un inicio de sesión de usuario en el gestor de colas ha fallado porque, por ejemplo, el ID de usuario del trabajo no es válido o adecuado, o el ID de usuario de la tarea o el ID de usuario alternativo no es válido. Uno o más de estos ID de usuario pueden no ser válidos porque se han revocado o suprimido. En este caso, recibe un mensaje ICHxxxx y, posiblemente, un mensaje IRRxxxx en el registro de trabajo del gestor de colas que explican la razón de la anomalía de inicio de sesión. Por ejemplo:

```
ICH408I USER(NOTDFND ) GROUP(        ) NAME(???                 )
  LOGON/JOB INITIATION - USER AT TERMINAL          NOT RACF-DEFINED
IRR012I  VERIFICATION FAILED. USER PROFILE NOT FOUND
```

- Se ha solicitado un usuario alternativo, pero el ID de usuario del trabajo o la tarea no tiene acceso al ID de usuario alternativo. Para esta anomalía, recibe un mensaje de violación en el registro de trabajo del gestor de colas correspondiente.
- Se ha utilizado una opción de contexto o está implícita en la apertura de una cola de transmisión para salida, pero el ID de usuario del trabajo o, según sea el caso, el ID de usuario alternativo o de la tarea no tiene acceso a la opción de contexto. En este caso, se pone un mensaje de violación en el registro de trabajo del gestor de colas correspondiente.
- Un usuario no autorizado ha intentado acceder a un objeto de gestor de colas protegido, por ejemplo, una cola. En este caso, se pone un mensaje ICH408I para la violación en el registro de trabajo del gestor de colas correspondiente. Esta violación puede deberse al ID de usuario del trabajo o, según sea el caso, al ID de usuario alternativo o de la tarea.

También pueden encontrarse mensajes de violación para la seguridad de mandatos y la seguridad de recursos de mandatos en el registro de trabajo del gestor de colas.

Si el mensaje de violación ICH408I muestra el nombre de trabajo del gestor de colas en lugar de un ID de usuario, esto suele ser el resultado de especificar un ID de usuario alternativo en blanco. Por ejemplo:

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
          MQS1.PAYROLL.REQUEST CL(MQQUEUE)
          INSUFFICIENT ACCESS AUTHORITY
          ACCESS INTENT(UPDATE )  ACCESS ALLOWED(NONE   )
```

Para averiguar quién tiene autorización para utilizar identificadores (ID) de usuario alternativo en blanco, compruebe la lista de acceso del perfil MQADMIN hlq.ALTERNATE.USER.-BLANK-.

También se puede generar un mensaje de violación ICH408I en las siguientes situaciones:

- Si se envía un mandato a la cola de entrada de mandatos del sistema sin contexto. Los programas escritos por el usuario que graban en la cola de entrada de mandatos del sistema deben utilizar siempre una opción de contexto. Para obtener más información, consulte “Perfiles para la seguridad de contexto” en la página 218.
- Cuando el trabajo que accede al recurso de IBM MQ no tiene asociado un ID de usuario, o cuando un adaptador de IBM MQ no puede extraer el ID de usuario del entorno del adaptador.

Es posible que también se emitan mensajes de violación si está utilizando seguridad a nivel de grupo de compartición de colas y a nivel de gestor de colas. Aunque puede recibir mensajes que indiquen que no se ha encontrado ningún perfil a nivel de gestor de colas, se le seguirá otorgando acceso debido a un perfil a nivel de grupo de compartición de colas.

```
ICH408I JOB(MQS1MSTR) STEP(MQS1MSTR)
          MQS1.PAYROLL.REQUEST CL(MQQUEUE)
          PROFILE NOT FOUND - REQUIRED FOR AUTHORITY CHECKING
          ACCESS INTENT(UPDATE )  ACCESS ALLOWED(NONE   )
```

z/OS Qué hacer si el acceso se ha otorgado o prohibido de forma incorrecta

Además de los pasos que se detallan en la publicación *z/OS Security Server RACF Security Administrator's Guide*, utilice esta lista de comprobación si el acceso a un recurso parece estar controlado de forma incorrecta.

- ¿Están los perfiles de conmutador establecidos correctamente?
  - ¿Está activo RACF?
  - ¿Están instaladas y activas las clases IBM MQ RACF?

    Utilice el mandato RACF SETROPTS LIST para comprobarlo.
  - Utilice el mandato de IBM MQ DISPLAY SECURITY para visualizar el estado actual de conmutador del gestor de colas.
  - Compruebe los perfiles de conmutador en la clase MQADMIN.

    Utilice RACF SEARCH y RLIST para hacer esto.
  - Vuelva a comprobar los perfiles de conmutador RACF emitiendo el mandato de IBM MQ REFRESH SECURITY(MQADMIN).
- ¿Ha cambiado el perfil de recurso RACF? Por ejemplo, ¿el acceso universal del perfil ha cambiado o la lista de acceso del perfil ha cambiado?
  - ¿El perfil es genérico?

    Si es así, emita el mandato RACF SETROPTS GENERIC(nombreclasse) REFRESH.
  - ¿Ha renovado la seguridad en este gestor de colas?

    Si es necesario, emita el mandato RACF SETROPTS RACLIST(nombreclase) REFRESH.

    Si es necesario, emita el mandato IBM MQ REFRESH SECURITY(*).
- ¿Ha cambiado la definición RACF del usuario? Por ejemplo, ¿se ha conectado al usuario a un nuevo grupo o se ha revocado la autorización de acceso del usuario?
  - ¿Ha verificado de nuevo al usuario emitiendo el mandato de IBM MQ RVERIFY SECURITY(idusuario)?
- ¿Se están pasando por alto las comprobaciones de seguridad debido a RESLEVEL?
  - Compruebe el acceso del ID de usuario de conexión al perfil RESLEVEL. Utilice los registros de auditoría RACF para determinar el valor en qué está establecido RESLEVEL.
  - Para los canales, recuerde que el nivel de acceso a RESLEVEL que tiene el ID de usuario del iniciador de canal lo heredan todos los canales, por lo que un nivel de acceso, como por ejemplo ALTER, que hace que se pasen por alto todas las comprobaciones hace que se pasen por alto todas las comprobaciones de seguridad para todos los canales.
  - Si está ejecutando desde CICS, compruebe el valor RESSEC de la transacción.
  - Si RESLEVEL se ha cambiado mientras un usuario estaba conectado, éste debe desconectarse y conectarse de nuevo para que el cambio tenga efecto.
- ¿Está utilizando grupos de compartición de colas?
  - Si está utilizando seguridad a nivel tanto de grupo de compartición de colas como de gestor de colas, compruebe que ha definido todos los perfiles correctos. Si el perfil de gestor de colas no se ha definido, se envía un mensaje al registro para indicar que no se ha encontrado el perfil.
  - ¿Ha utilizado una combinación de valores de conmutador que no es válida de manera que se ha activado la comprobación de seguridad completa?
  - ¿Necesita definir conmutadores de seguridad para alterar temporalmente algunos de los valores de grupo de compartición de colas para el gestor de colas?
  - ¿Hay un perfil a nivel de gestor de colas que está teniendo prioridad sobre un perfil a nivel de grupo de compartición de colas?

z/OS Consideraciones de seguridad para el iniciador de canal en z/OS

Si está utilizando seguridad de recursos en un entorno de gestión de colas distribuidas, el espacio de direcciones del iniciador de canal necesita acceso adecuado a diversos recursos de IBM MQ. Puede utilizar Integrated Cryptographic Support Facility (ICSF) para inicializar el algoritmo de protección por contraseña.

Utilización de seguridad de recursos

Si está utilizando seguridad de recursos, tenga en cuenta los puntos siguientes si está utilizando la gestión de colas distribuidas:

Colas del sistema
El espacio de direcciones del iniciador de canal necesita acceso RACF UPDATE a las colas del sistema que se listan en "Seguridad de colas del sistema" en la página 207, y a todas las colas de destino de usuario y la cola de mensajes no entregados (pero consulte "Seguridad de la cola de mensajes no entregados" en la página 206).

Colas de transmisión
El espacio de direcciones del iniciador de canal necesita acceso ALTER a todas las colas de transmisión de usuario.

Seguridad de contexto
El ID de usuario de canal (y el ID de usuario de MCA, si se ha especificado uno) necesitan acceso RACF CONTROL a los perfiles hlq.CONTEXT.queuename de la clase MQADMIN. En función del perfil RESLEVEL, puede que el ID de usuario del canal necesite también acceso CONTROL a estos perfiles.

Todos los canales necesitan acceso CONTROL a MQADMIN hlq.CONTEXT. perfil de cola de mensajes no entregados. Todos los canales (ya sean de inicio o de respuesta) pueden generar informes y, por consiguiente, necesitan acceso CONTROL al perfil hlq.CONTEXT.reply-q.

Los canales SENDER, CLUSSDR y SERVER necesitan acceso CONTROL a los perfiles hlq.CONTEXT.nombre-cola-transmisión ya que se pueden transferir mensajes a la cola de transmisión para "despertar" al canal para que finalice de forma ordenada.

Nota: Si el ID de usuario de canal, o un grupo RACF al que el ID de usuario de canal está conectado, tiene acceso CONTROL o ALTER al perfil hlq.RESLEVEL, entonces no hay comprobaciones de recursos para el iniciador de canal ni para ninguno de sus canales.

En el apartado "Perfiles para la seguridad de contexto" en la página 218 "RESLEVEL y la conexión del iniciador de canal" en la página 238 y "ID de usuario para la comprobación de seguridad en z/OS" en la página 240 hallará más información.

CSQINPX
Si está utilizando el conjunto de datos de entrada CSQINPX, el iniciador de canal también necesita acceso READ a CSQINPX, y acceso UPDATE al conjunto de datos CSQOUTX y a las colas dinámicas SYSTEM.CSQXCMD.*.

Seguridad de conexión
Las solicitudes de conexión del espacio de direcciones del iniciador de canal utilizan un tipo de conexión CHIN, para el que debe establecerse la seguridad de acceso adecuada; consulte "Perfiles de seguridad de conexión para el iniciador de canal" en la página 200.

Conjuntos de datos
El espacio de direcciones del iniciador de canal necesita acceso adecuado a los conjuntos de datos del gestor de colas, consulte "Autorizar el acceso a los conjuntos de datos" en la página 257.

Mandatos
Los mandatos de gestión de colas distribuidas (por ejemplo, DEFINE CHANNEL, START CHINIT, START LISTENER y otros mandatos de canal) deben tener establecida la seguridad de mandatos adecuada; consulte la Tabla 49 en la página 221.

Si está utilizando un grupo de compartición de colas, el iniciador de canal puede emitir varios mandatos internamente, por lo que el ID de usuario que éste utiliza debe estar autorizado para

emitir dichos mandatos. Estos mandatos son START y STOP CHANNEL para cada canal utilizado con CHLDISP(SHARED).

Si PSMODE del gestor de colas no está inhabilitado (DISABLED), el iniciador de canal debe tener acceso de lectura (READ) para el mandato DISPLAY PUBSUB.

Seguridad de canal

Los canales, especialmente los receptores y los de conexión con el servidor, necesitan tener configurada la seguridad adecuada; consulte "ID de usuario para la comprobación de seguridad en z/OS" en la página 240 para obtener más información.

También puede utilizar el protocolo TLS (seguridad de la capa de transporte) para proporcionar seguridad en los canales. Consulte "Protocolos de seguridad TLS en IBM MQ" en la página 24 si desea más información sobre el uso de TLS con IBM MQ.

Consulte también "Control de accesos para clientes" en la página 98 para obtener información sobre la seguridad de la conexión con el servidor.

ID de usuario

Los ID de usuario descritos en "Identificadores de usuario utilizados por el iniciador de canal" en la página 243 e "Identificadores de usuario utilizados por el agente de transferencia a colas entre grupos" en la página 248 necesitan el acceso siguiente:

- Acceso RACF UPDATE a las colas de destino adecuadas y a la cola de mensajes no entregados
- Acceso RACF CONTROL al perfil `hlq.CONTEXT.queuename` si se realiza comprobación de contexto en el receptor
- Acceso adecuado a los perfiles hlq.ALTERNATE.USER.userid que puedan tener que utilizar.
- Para los clientes, el acceso RACF adecuado a los recursos que se van a utilizar.

Seguridad APPC

Establezca la seguridad APPC adecuada si está utilizando el protocolo de transmisión LU 6.2. (Por ejemplo, utilice la clase APPCLU de RACF.) Para obtener información sobre la configuración de la seguridad para APPC, consulte los siguientes manuales:

- *z/OS V1R2.0 Planificación de MVS: Gestión de APPC*
- *Multiplatform APPC Configuration Guide*, una publicación IBM Redbooks

Las transmisiones de salida utilizan la opción APPC "SECURITY(SAME)". Como resultado, el ID de usuario del espacio de direcciones del iniciador de canal y su perfil predeterminado (RACF GROUP) se transmiten a través de la red al receptor con un indicador de que el ID de usuario ya se ha verificado (ALREADYV).

Si el área de recepción también es z/OS, el ID de usuario y el perfil son verificados por APPC y el ID de usuario es presentado al canal receptor y se utiliza como el ID de usuario de canal.

En un entorno en el que el gestor de colas está utilizando APPC para comunicarse con otro gestor de colas del mismo o de otro sistema z/OS, necesita asegurarse de que:

- La definición VTAM para la LU de comunicación especifica SETACPT(ALREADYV)
- Hay un perfil RACF APPCLU para la conexión entre las LU que especifica CONVSEC(ALREADYV)

Cambiar valores de seguridad

Si se cambia el nivel de acceso RACF que el ID de usuario de canal o el ID de usuario de MCA tiene a una cola de destino, este cambio sólo tiene efecto para los nuevos manejadores de objeto (es decir, nuevas MQOPEN) para la cola de destino. El momento en que los MCA abren y cierran colas es variable; si un canal ya está ejecutándose cuando se realiza un cambio de acceso de este tipo, el MCA puede seguir transfiriendo mensajes a la cola de destino utilizando el acceso de seguridad existente de los ID de usuario, en lugar del acceso de seguridad actualizado. La detención y reinicio de los canales para implementar el nivel de acceso actualizado evita esta situación.

Reinicio automático
Si está utilizando el gestor de reinicio automático (Automatic Restart Manager - ARM) de z/OS para reiniciar el iniciador de canal, el ID de usuario asociado al espacio de direcciones XCFAS debe estar autorizado para emitir el mandato de IBM MQ MQ START CHINIT.

Utilización de Integrated Cryptographic Service Facility (ICSF)

El iniciador de canal puede utilizar ICSF para generar un número aleatorio cuando se inicializa el algoritmo de protección por contraseña para enmascarar contraseñas que se envían a canales de cliente si no se está utilizando TLS. El proceso de generar un número aleatorio se denomina *entropy*.

Si tiene la característica z/OS instalada pero no ha iniciado ICSF, verá el mensaje CSQX213E y el iniciador de canal utiliza STCK para entropía.

El mensaje CSQX213E le avisa que el algoritmo de protección por contraseña no es lo seguro que debería ser. No obstante, puede seguir configurando el proceso; no hay ningún otro impacto en el tiempo de ejecución.

Si no tiene instalada la característica z/OS, el iniciador de canal utiliza automáticamente STCK.

Notas:

1. Si se utiliza ICSF para la entropía se generan más secuencias que si se utiliza STCK.
2. Si inicia ICSF debe reiniciar el iniciador de canal.
3. ICSF es necesario para determinadas CipherSpecs. Si intenta utilizar una de estas CipherSpecs y no tiene instalado ICSF, recibirá el mensaje CSQX629E.

z/OS Seguridad de los clústeres de gestores de colas en z/OS

Las consideraciones de seguridad para los clústeres son las mismas que para los gestores de colas y canales que no están agrupados en clúster. El iniciador de canal necesita tener acceso a algunas colas del sistema adicionales, y algunos mandatos adicionales necesitan tener establecida la seguridad adecuada.

Puede utilizar el ID de usuario de MCA, los registros de autenticación de canal, TLS, y las salidas de seguridad para autenticar a los canales de clúster (como con los canales convencionales). Los registros de autenticación de canal o la salida de seguridad relacionados con el canal de clúster receptor deben comprobar que el gestor de colas tenga permiso para acceder a los clústeres del gestor de colas del servidor. Puede empezar a utilizar el soporte de clúster de IBM MQ sin tener que cambiar la seguridad de acceso a colas existente. Aunque debe permitir a otros gestores de colas del clúster grabar en la cola SYSTEM.CLUSTER.COMMAND.QUEUE si van a unirse al clúster.

El soporte de clúster IBM MQ no proporciona un mecanismo para limitar a un miembro de un clúster sólo al rol de cliente. Como resultado, debe estar seguro de que confía en los gestores de colas a los que permite unirse al clúster. Si algún gestor de colas del clúster crea una cola con un nombre determinado, éste puede recibir mensajes para esa cola, independientemente de si la aplicación que transfiere mensajes a esa cola tenía esta intención o no.

Para restringir la pertenencia a un clúster, tome las mismas medidas que tomaría para impedir que los gestores de colas se conecten a canales receptores. La pertenencia a un clúster se restringe utilizando los registros de autenticación de canal o grabando un programa de salida de seguridad en el canal receptor. También puede escribir un programa de salida que impida que los gestores de colas no autorizados escriban en SYSTEM.CLUSTER.COMMAND.QUEUE.

Nota: No es aconsejable permitir que las aplicaciones abran la cola SYSTEM.CLUSTER.TRANSMIT.QUEUE directamente. Tampoco es aconsejable permitir que una aplicación abra directamente cualquier otra cola de transmisión.

Si está utilizando seguridad de recursos, tenga en cuenta los puntos siguientes además de las consideraciones descritas en “Consideraciones de seguridad para el iniciador de canal en z/OS” en la página 265:

Colas del sistema
El iniciador de canal necesita acceso RACF ALTER a las siguientes colas de sistema:

- SYSTEM.CLUSTER.COMMAND QUEUE
- SYSTEM.CLUSTER.TRANSMIT.QUEUE.

y acceso UPDATE a SYSTEM.CLUSTER.REPOSITORY.QUEUE

También necesita acceso READ a cualquier lista de nombres que se utilice para la agrupación en clúster.

Mandatos
Establezca la seguridad de mandatos adecuada (como se describe en la Tabla 49 en la página 221) para los mandatos de soporte de clúster (REFRESH y RESET CLUSTER, SUSPEND y RESUME QMGR).

z/OS Consideraciones de seguridad para el uso de IBM MQ con CICS

Todas las versiones de CICS soportadas por IBM MQ 9.0.0 y posteriores, utilizan la versión suministrada por CICS del adaptador y el puente.

Para obtener los detalles sobre las consideraciones de seguridad, consulte:

- Seguridad del adaptador CICS-IBM MQ.
- Seguridad para el puente CICS-IBM MQ.

z/OS Consideraciones de seguridad para el uso de IBM MQ con IMS

Utilice este tema para planificar sus requisitos de seguridad cuando utilice IBM MQ con IMS.

Utilización de la clase OPERCMDS

Si utiliza RACF para proteger recursos de la clase OPERCMDS, asegúrese de que el ID de usuario asociado al espacio de direcciones del gestor de colas de IBM MQ tenga autorización para emitir el mandato MODIFY para cualquier sistema IMS al que se pueda conectar.

Consideraciones de seguridad para el puente IMS

Hay cuatro aspectos que debe tener en cuenta al decidir los requisitos de seguridad para el puente IMS. Estos son:

- Qué autorización de seguridad se necesita para conectar IBM MQ a IMS
- Qué grado de comprobación de seguridad se realiza en las aplicaciones que utilizan el puente para acceder a IMS
- Qué recursos de IMS están autorizados a utilizar estas aplicaciones
- Qué autorización se va a utilizar para los mensajes transferidos y obtenidos por el puente

Cuando defina los requisitos de seguridad para el puente IMS, debe tener en cuenta lo siguiente:

- Los mensajes que pasan a través del puente se pueden haber originado en aplicaciones de plataformas que no ofrecen características de seguridad potentes
- Los mensajes que pasan a través del puente se pueden haber originado en aplicaciones que no están controladas por la misma empresa u organización

z/OS *Consideraciones de seguridad para conectarse a IMS*

Otorgue al ID de usuario del espacio de direcciones del gestor de colas IBM MQ acceso al grupo OTMA.

El puente IMS es un cliente OTMA. La conexión a IMS funciona bajo el ID de usuario del espacio de direcciones del gestor de colas IBM MQ. Éste se define normalmente como miembro del grupo de tareas iniciadas. Debe otorgarse a este ID de usuario acceso al grupo OTMA (a menos que el valor de /SECURE OTMA sea NONE).

Para ello, defina el siguiente perfil en la clase FACILITY:

```
IMSXCF.xcfgname.mqxcfmname
```

Donde `xcfgname` es el nombre de grupo XCF e `mqxcfmname` es el nombre de miembro XCF para IBM MQ.

Debe otorgar al ID de usuario del gestor de colas IBM MQ acceso de lectura a este perfil.

Nota:

1. Si cambia las autorizaciones en la clase FACILITY, debe emitir el mandato RACF SETROPTS RACLIST(FACILITY) REFRESH para activar los cambios.
2. Si el perfil hlq.NO.SUBSYS.SECURITY existe en la clase MQADMIN, no se pasa ningún ID de usuario a IMS y la conexión falla a menos que el valor de /SECURE OTMA sea NONE.

z/OS *Control de acceso a aplicaciones para el puente IMS*

Defina un perfil RACF en la clase FACILITY para cada sistema IMS. Otorgue un nivel de acceso adecuado al ID de usuario del gestor de colas de IBM MQ.

Para cada sistema IMS al que se conecta el puente IMS, puede definir el siguiente perfil RACF en la clase FACILITY para determinar qué grado de comprobación de seguridad se realiza para cada mensaje que se pasa al sistema IMS.

```
IMSXCF.xcfgname.imsxcfmname
```

Donde `xcfgname` es el nombre de grupo XCF e `imsxcfmname` es el nombre de miembro XCF para IMS. (Debe definir un perfil aparte para cada sistema IMS.)

El nivel de acceso que permita para el ID de usuario del gestor de colas IBM MQ en este perfil se devuelve a IBM MQ cuando el puente IMS se conecta a IMS, e indica el nivel de seguridad que se requiere en transacciones posteriores. Para las transacciones posteriores, IBM MQ solicita los servicios adecuados desde RACF y, si el ID de usuario tiene autorización, pasa el mensaje a IMS.

OTMA no da soporte al mandato IMS /SIGN; no obstante, IBM MQ le permite establecer la comprobación de acceso para cada mensaje, a fin de permitir la implementación del nivel de control necesario.

Se puede devolver la siguiente información de nivel de acceso:

NONE o NO PROFILE FOUND
: Estos valores indican que se requiere seguridad máxima, es decir, se requiere autenticación para cada transacción. Se realiza una comprobación para verificar que el ID de usuario especificado en el campo *IdentificadorUsuario* de la estructura MQMD, y la contraseña o PassTicket en el campo *Autenticador* de la estructura MQIIH, son reconocidos por RACF y son una combinación válida. Se crea una UTOKEN con una contraseña o PassTicket, y se pasa a IMS; la UTOKEN no se almacena en memoria caché.

  Nota: Si el perfil hlq.NO.SUBSYS.SECURITY existe en la clase MQADMIN, este nivel de seguridad altera temporalmente todo lo que está definido en el perfil.

READ
: Este valor indica que se debe realizar la misma autenticación que para `NONE` en las siguientes circunstancias:

  - La primera vez que se encuentra un ID de usuario específico
  - Cuando el ID de usuario se ha encontrado antes pero la UTOKEN almacenada en memoria caché no se ha creado con una contraseña o PassTicket

  IBM MQ solicita una UTOKEN si es necesario, y la pasa a IMS.

Nota: Si se ha obedecido una solicitud para volver a verificar la seguridad, se pierde toda la información almacenada en la memoria caché y se solicita una UTOKEN la primera vez que se encuentra posteriormente cada ID de usuario.

UPDATE
Se realiza una comprobación de que el ID de usuario en el campo *UserIdentifier* de la estructura MQMD es reconocido por RACF.

Se crea una UTOKEN y se pasa a IMS; la UTOKEN se almacena en memoria caché.

CONTROL/ALTER
Estos valores indican que no es necesario proporcionar ninguna UTOKEN de seguridad para ningún ID de usuario para este sistema IMS. (Es probable que sólo utilice esta opción para sistemas de desarrollo y de prueba.)

Atención: Tenga en cuenta que el ID de usuario incluido en el campo *UserIdentifier* de la estructura MQMD se sigue pasando para **CONTROL/ALTER**.

Nota:

1. Este acceso se define cuando IBM MQ se conecta a IMS, y su duración es la misma que la de la conexión. Para cambiar el nivel de seguridad, se debe cambiar el acceso al perfil de seguridad y luego se debe detener y reiniciar el puente (por ejemplo, deteniendo y reiniciando OTMA).
2. Si cambia las autorizaciones en la clase FACILITY, debe emitir el mandato RACF SETROPTS RACLIST(FACILITY) REFRESH para activar los cambios.
3. Puede utilizar una contraseña o un Pase, pero debe tener presente que el puente IMS no cifra los datos. Para obtener información sobre el uso de Pases, consulte "Utilización de PassTickets RACF en la cabecera IMS" en la página 271.
4. Algunos de estos resultados podrían verse afectados por los valores de seguridad de IMS, mediante el mandato /SECURE OTMA.
5. La información UTOKEN almacenada en memoria caché se guarda durante el periodo de tiempo definido por los parámetros INTERVAL y TIMEOUT del mandato de IBM MQ ALTER SECURITY.
6. La opción RACF WARNING no tiene ningún efecto en el perfil IMSXCF.xcfgname.imsxcfmname. Su uso no afecta al nivel de acceso otorgado y no se producen mensajes RACF WARNING.

z/OS *Comprobación de seguridad en IMS*

Los mensajes que pasan a través del puente contienen información de seguridad. Las comprobaciones de seguridad realizadas dependen del valor del mandato /SECURE OTMA de IMS.

Cada mensaje de IBM MQ que pasa a través del puente contiene la siguiente información de seguridad:

- Un ID de usuario incluido en el campo *IdentificadorUsuario* de la estructura MQMD
- El ámbito de seguridad incluido en el campo *ÁmbitoSeguridad* de la estructura MQIIH (si la estructura MQIIH está presente)
- Una UTOKEN (a menos que el subsistema IBM MQ tenga accesos CONTROL o ALTER al perfil `IMSXCF.xcfgname.imsxcfmname` pertinente)

Las comprobaciones de seguridad que se realizan dependen del valor del mandato /SECURE OTMA de IMS, como se indica a continuación:

/SECURE OTMA NONE
No se realiza ninguna comprobación de seguridad para la transacción.

/SECURE OTMA CHECK
El campo *IdentificadorUsuario* de la estructura MQMD se pasa a IMS para la comprobación de autorización de la transacción o mandato.

Se construye un ACEE (Accessor Environment Element) en la región de control de IMS.

/SECURE OTMA FULL
El campo *IdentificadorUsuario* de la estructura MQMD se pasa a IMS para la comprobación de autorización de la transacción o mandato.

Se construye un ACEE en la región dependiente de IMS y en la región de control de IMS.

/SECURE OTMA PROFILE
El campo *IdentificadorUsuario* de la estructura MQMD se pasa a IMS para la comprobación de autorizaciones de transacción o de mandatos

El campo *ÁmbitoSeguridad* de la estructura MQIIH se utiliza para determinar si se debe construir un ACEE en la región dependiente de IMS, además de en la región de control.

Nota:

1. Si cambia las autorizaciones en la clase TIMS o CIMS, o las clases de grupo asociado GIMS o DIMS, debe emitir los siguientes mandatos IMS para activar los cambios:
   - /MODIFY PREPARE RACF
   - /MODIFY COMMIT
2. Si no utiliza /SECURE OTMA PROFILE, se hace caso omiso de cualquier valor especificado en el campo *ÁmbitoSeguridad* de la estructura MQIIH.

z/OS *Comprobación de seguridad realizada por el puente IMS*

Se utilizan autorizaciones diferentes en función de la acción que se realiza.

Cuando el puente transfiere u obtiene un mensaje, se utilizan las siguientes autorizaciones:

Obtener un mensaje de la cola puente
No se realiza ninguna comprobación de seguridad.

Transferir una excepción o un mensaje de informe COA
Utiliza la autorización del ID de usuario existente en el campo *IdentificadorUsuario* de la estructura MQMD.

Transferir un mensaje de respuesta
Utiliza la autorización del ID de usuario existente en el campo *IdentificadorUsuario* de la estructura MQMD del mensaje original.

Transferir un mensaje a la cola de mensajes no entregados
No se realiza ninguna comprobación de seguridad.

Nota:

1. Si cambia los perfiles de clase de IBM MQ, debe emitir el mandato de IBM MQ REFRESH SECURITY(*) para activar los cambios.
2. Si cambia la autorización de un usuario, debe emitir el mandato MQSC RVERIFY SECURITY para activar el cambio.

z/OS *Utilización de PassTickets RACF en la cabecera IMS*

Puede utilizar un PassTicket en lugar de una contraseña en la cabecera IMS.

Si desea utilizar un PassTicket en lugar de una contraseña en la cabecera IMS (MQIIH), especifique el nombre de la aplicación contra la que se validará el PassTicket en el atributo PASSTKTA de la definición STGCLASS de la cola Puente de IMS a la que se direccionará el mensaje.

Si el valor PASSTKTA se deja en blanco, debe disponer lo necesario para que se genere un PassTicket. En este caso, el nombre de aplicación debe tener el formato MVSxxxx, donde xxxx es el SMFID del sistema z/OS en el que se ejecuta el gestor de colas de destino.

Un PassTicket se construye a partir de un ID de usuario, el nombre de la aplicación de destino y una clave secreta. Es un valor de 8 bytes que contiene caracteres alfabéticos en mayúsculas y numéricos. Se puede utilizar una sola vez y es válido durante un periodo de 20 minutos. Si el PassTicket lo genera un sistema RACF local, RACF sólo comprueba si el perfil existe y no si el usuario tiene autorización para el perfil. Si el PassTicket se ha generado en un sistema remoto, RACF valida el acceso del ID de usuario al perfil. Para obtener información completa acerca de PassTickets, consulte la publicación *z/OS SecureWay Security Server RACF Security Administrator's Guide*.

Los PassTickets de las cabeceras IMS se proporcionan a RACF mediante IBM MQ, no IMS.

z/OS z/OS Migración de un gestor de colas a la seguridad de mayúsculas y minúsculas

Siga estos pasos para migrar un gestor de colas a una seguridad de mayúsculas y minúsculas. Revise el nivel de producto de seguridad que está utilizando y active las nuevas clases del supervisor de seguridad externo de IBM MQ. Ejecute el mandato **REFRESH SECURITY** para activar los perfiles con mayúsculas y minúsculas.

Antes de empezar

1. Asegúrese de que están activadas todas las clases de supervisor de seguridad externo de IBM MQ.
2. Asegúrese de que el gestor de colas se ha iniciado.

Acerca de esta tarea

Siga estos pasos para convertir un gestor de colas a una seguridad de mayúsculas y minúsculas.

Procedimiento

1. Copie todos los perfiles y niveles de acceso existentes de las clases en mayúsculas en la clase del supervisor de seguridad externo con mayúsculas y minúsculas equivalente.
   a) `MQADMIN` por `MXADMIN`.
   b) `MQPROC` por `MXPROC`.
   c) `MQNLIST` por `MXNLIST`.
   d) `MQQUEUE` por `MXQUEUE`.
2. Cambie el valor del atributo `SCYCASE` por `MIXED`.

```
ALTER QMGR SCYCASE(MIXED)
```

3. Active los perfiles de seguridad existentes.

```
REFRESH SECURITY(*) TYPE(CLASSES)
```

4. Compruebe que los perfiles de seguridad funcionan correctamente.

Qué hacer a continuación

Revise las definiciones de objeto y cree nuevos perfiles de mayúsculas/minúsculas como sea adecuado, utilizando **REFRESH SECURITY** como necesario para activar los perfiles.

Configuración de la seguridad de IBM MQ MQI client

Debe tener en cuenta la seguridad del IBM MQ MQI client para que las aplicaciones cliente no tengan acceso sin restricciones a los recursos del servidor.

Al ejecutar una aplicación cliente, no ejecute la aplicación utilizando un ID de usuario que tenga más derechos de acceso que los necesarios; por ejemplo, un usuario en el grupo `mqm` o incluso el propio usuario `mqm`.

Al ejecutar una aplicación como un usuario con demasiados derechos de acceso, corre el riesgo de que la aplicación acceda y cambie partes del gestor de colas, ya sea de forma accidental o intencional.

Hay dos aspectos de seguridad entre una aplicación cliente y su servidor de gestor de colas: la autenticación y el control de accesos.

- La autenticación puede utilizarse para asegurarse de que la aplicación cliente, ejecutándose como un usuario específicos, sea quien dice ser. Utilizando la autenticación podrá impedir que un atacante acceda al gestor de colas suplantando una de sus aplicaciones:

  A partir de IBM MQ 8.0, la autenticación se proporciona mediante una de dos opciones:

  – La característica de autenticación de conexión.

    Para obtener más información sobre la autenticación de conexión, consulte "Autenticación de conexión" en la página 67.

  – Utilización de la autenticación mutua dentro de TLS.

    Para obtener más información sobre TLS, consulte "Trabajar con SSL/TLS" en la página 277.

- El control de acceso puede utilizarse para otorgar o eliminar derechos de acceso para un usuario específico o un grupo de usuarios. Si ejecuta una aplicación cliente con un usuario creado de forma específica (o usuario en un grupo específico) podrá utilizar los controles de acceso para asegurarse de que la aplicación no pueda acceder a partes del gestor de colas que la aplicación no debería.

  Al configurar el control de acceso deberá tener en cuenta las reglas de autenticación de canal y el campo MCAUSER en un canal. Ambas características tienen la capacidad de cambiar qué ID de usuario se está utilizando para verificar derechos de control de acceso.

  Para obtener más información sobre el control de acceso, consulte "Autorización del acceso a objetos" en la página 355.

Si ha configurado una aplicación cliente para conectarse a un canal específico con un ID restringido, pero el canal tiene un ID de administrador establecido en el campo MCAUSER, siempre y cuando la aplicación cliente se conecte satisfactoriamente, el ID de administrador se utilizará para las comprobaciones de control de acceso. Por lo tanto, la aplicación cliente tendrá derechos de acceso total al gestor de colas.

Para obtener más información sobre el atributo MCAUSER, consulte "Correlación de un ID de usuario cliente con un ID de usuario MCAUSER" en la página 394.

Las reglas de autenticación de canal también pueden utilizarse como método para controlar el acceso a un gestor de colas, estableciendo criterios y reglas específicas para que se acepte una conexión.

Para obtener más información sobre las reglas de autenticación de canal, consulte "Registros de autenticación de canal" en la página 49.

Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI

Cree sus propios repositorios de claves utilizando software compatible con FIPS y, a continuación, especifique que el canal debe utilizar las CipherSpecs certificadas por FIPS.

Para que sean compatibles con FIPS en el tiempo de ejecución, los repositorios de claves se deben haber creado y gestionado utilizando software compatible sólo con FIPS como, por ejemplo, runmqakm, con la opción -fips.

Puede especificar que un canal TLS debe utilizar sólo CipherSpecs certificadas por FIPS de tres maneras, listadas por orden de prioridad:

1. Establezca el campo FipsRequired de la estructura MQSCO en MQSSL_FIPS_YES.
2. Establezca la variable de entorno MQSSLFIPS en YES.
3. Establezca el atributo SSLFipsRequired en el archivo de configuración de cliente en YES.

De forma predeterminada, las CipherSpecs certificadas por FIPS no son obligatorias.

Estos valores tienen los mismos significados que los valores de los parámetros equivalentes en ALTER QMGR SSLFIPS (consulte ALTER QMGR). Si el proceso de cliente no tiene actualmente ninguna conexión TLS activa y se especifica un valor FipsRequired válido en una llamada MQCONNX de SSL, todas las conexiones TLS posteriores asociadas a este proceso deben utilizar únicamente las CipherSpecs asociadas a este valor. Esto se aplica hasta que ésta y el resto de conexiones TLS se hayan detenido, momento en el que una MQCONNX posterior puede proporcionar un nuevo valor para FipsRequired.

Si el hardware de cifrado está configurado, los módulos de cifrado que IBM MQ utiliza se pueden configurar con los módulos que proporciona el producto de hardware y estos pueden estar certificados por FIPS en un nivel determinado. Los módulos configurables y si tienen el certificado FIPS depende del producto de hardware que se utilice.

Cuando sea posible, si están configuradas las CipherSpecs sólo de FIPS, el cliente de MQI rechaza conexiones que especifiquen una CipherSpec que no sea FIPS con MQRC_SSL_INITIALIZATION_ERROR. IBM MQ no garantiza rechazar todas las conexiones de este tipo y es responsabilidad del usuario determinar si la configuración es compatible con IBM MQ.

Conceptos relacionados

"Federal Information Processing Standards (FIPS) para UNIX, Linux, and Windows" en la página 33
Cuando se requiere criptografía en un canal SSL/TLS en sistemas Windows, UNIX and Linux, IBM MQ utiliza un paquete de criptografía denominado IBM Crypto for C (ICC). En las plataformas Windows, UNIX and Linux, , el software ICC ha pasado el programa de validación de módulo de cifrado de FIPS (estándares federales de procesamiento de la información) del Instituto Nacional de Estándares y Tecnología, en el nivel 140-2.

Stanza SSL del archivo de configuración de cliente

Referencia relacionada

FipsRequired (MQLONG)

MQSSLFIPS

AIX Ejecución de aplicaciones cliente TLS con varias instalaciones de GSKit V8.0 en AIX

Las aplicaciones cliente TLS en AIX pueden experimentar un error MQRC_CHANNEL_CONFIG_ERROR y AMQ6175 cuando se ejecutan en sistemas AIX con varias instalaciones GSKit V8.0.

Al ejecutar aplicaciones cliente en un sistema AIX con varias instalaciones GSKit V8.0, las llamadas de conexión de cliente pueden devolver MQRC_CHANNEL_CONFIG_ERROR al utilizar TLS. Los registros de `/var/mqm/errors` registran el error AMQ6175 y AMQ9220 para la aplicación cliente anómala, por ejemplo:

```
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 ------------------------------------------------------
```

Una causa común de este error es que el valor de la variable de entorno LIBPATH o LD_LIBRARY_PATH ha hecho que el cliente IBM MQ cargue un conjunto mixto de bibliotecas de dos instalaciones de GSKit V8.0 diferentes. La ejecución de una aplicación cliente IBM MQ en un entorno de Db2 puede producir este error.

Para evitarlo, incluya los directorios de la biblioteca de IBM MQ en la parte frontal de la vía de acceso de bibliotecas para que las bibliotecas de IBM MQ tengan prioridad. Esto se puede lograr utilizando el mandato **setmqenv** con el parámetro **-k** , por ejemplo:

```
. /usr/mqm/bin/setmqenv -s -k
```

Para obtener más información sobre el uso del mandato **setmqenv**, consulte setmqenv (establezca el entorno de IBM MQ)

IBM i Configuración de las comunicaciones para SSL o TLS en IBM i

Las comunicaciones seguras que utilizan los protocolos de seguridad de cifrado SSL o TLS comportan la configuración de canales de comunicación y la gestión de los certificados digitales que utilizará para la autenticación.

Para configurar la instalación de SSL o TLS, debe definir los canales para que utilicen SSL o TLS. También debe crear y gestionar los certificados digitales. En algunos sistemas operativos, puede realizar las pruebas con certificados autofirmados. Sin embargo, en IBM i, debe utilizar certificados personales firmados por una CA local.

Para obtener información completa sobre la creación y gestión de certificados, consulte “Trabajar con SSL/TLS en IBM i” en la página 277.

En esta colección de temas se presentan algunas de las tareas que forman parte de la configuración de las comunicaciones SSL o TLS, y se proporciona una guía paso a paso para completar estas tareas.

Es posible que también desee probar la autenticación de cliente SSL o TLS, que son partes opcionales de los protocolos SSL y TLS. Durante el reconocimiento SSL o TLS, el cliente TLS o SSL siempre obtiene y valida un certificado digital del servidor. Con la implementación de IBM MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En IBM i, el cliente SSL o TLS solo envía un certificado si tiene uno etiquetado con el formato de IBM MQ correcto:

- Para un gestor de colas, `ibmwebspheremq` seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`.
- Para un cliente C de IBM MQ para IBM i, `ibmwebspheremq` seguido del ID de usuario de inicio de sesión cambiado a minúsculas, por ejemplo, `ibmwebspheremqmyuserid`.

IBM MQ utiliza el prefijo `ibmwebspheremq` en una etiqueta para evitar confusiones con certificados de otros productos. Asegúrese de especificar la etiqueta completa del certificado en minúsculas.

El servidor SSL o TLS siempre valida el certificado de cliente si se envía uno. Si el cliente SSL o TLS no envía un certificado, la autenticación falla solo si el extremo del canal que actúa como servidor SSL o TLS se define con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información, consulte Conexión de dos gestores de colas utilizando SSL o TLS.

ULW Configuración de las comunicaciones para SSL o TLS en UNIX, Linux o Windows

Las comunicaciones seguras que utilizan los protocolos de seguridad de cifrado SSL o TLS comportan la configuración de canales de comunicación y la gestión de los certificados digitales que utilizará para la autenticación.

Para configurar la instalación de SSL o TLS, debe definir los canales para que utilicen SSL o TLS. También debe crear y gestionar los certificados digitales. En los sistemas UNIX, Linux y Windows, puede realizar las pruebas con certificados autofirmados.

Atención: No es posible utilizar una combinación de certificados firmados por Elliptic Curve y los certificados firmados por RSA en los gestores de colas que desea unir utilizando los canales habilitados para TLS.

Los gestores de colas que utilizan los canales habilitados para TLS deben utilizar todos los certificados firmados por RSA, o bien todos los certificados firmados por EC, no una combinación de ambos.

Consulte “Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45 para obtener más información.

Los certificados autofirmados no se pueden revocar, lo que podría permitir a un atacante suplantar una identidad después de que se haya comprometido una clave privada. Las CA pueden revocar un certificado comprometido, lo que impide su posterior uso. Los certificados firmados por una CA son, por lo tanto, más seguros para su uso en un entorno de producción, aunque los certificados autofirmados son más convenientes para un sistema de prueba.

Para obtener información completa sobre la creación y gestión de certificados, consulte “Trabajar con SSL/TLS en UNIX, Linux, and Windows” en la página 289.

En esta colección de temas se presentan algunas de las tareas que forman parte de la configuración de las comunicaciones SSL y se proporciona una guía paso a paso para completar estas tareas.

Es posible que también desee probar la autenticación de cliente SSL o TLS, que es una parte opcional de los protocolos. Durante el reconocimiento SSL o TLS, el cliente TLS o SSL siempre obtiene y valida un certificado digital del servidor. Con la implementación de IBM MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En UNIX, Linux, and Windows, el cliente SSL o TLS solo envía un certificado si tiene uno etiquetado con el formato de IBM MQ correcto:

- Para un gestor de colas, el formato es `ibmwebspheremq` seguido del nombre del gestor de colas que ha cambiado a minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`
- Para un cliente de IBM MQ , `ibmwebspheremq` seguido por el ID de usuario de inicio de sesión cambiado a minúsculas, por ejemplo, `ibmwebspheremqmyuserid`.

IBM MQ utiliza el prefijo `ibmwebspheremq` en una etiqueta para evitar confusiones con certificados de otros productos. Asegúrese de especificar la etiqueta completa del certificado en minúsculas.

El servidor SSL o TLS siempre valida el certificado de cliente si se envía uno. Si el cliente no envía un certificado, la autenticación falla sólo si el extremo del canal que actúa como servidor SSL o TLS se define con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información, consulte Conexión de dos gestores de colas utilizando SSL o TLS.

z/OS Configuración de las comunicaciones para SSL o TLS en z/OS

Las comunicaciones seguras que utilizan los protocolos de seguridad de cifrado SSL o TLS comportan la configuración de canales de comunicación y la gestión de los certificados digitales que utilizará para la autenticación.

Para configurar la instalación de SSL o TLS, debe definir los canales para que utilicen SSL o TLS. También debe crear y gestionar los certificados digitales. En z/OS, puede realizar las pruebas con certificados autofirmados, o con certificados personales firmados por una entidad emisora de certificados (CA) local.

Los certificados autofirmados no se pueden revocar, lo que podría permitir a un atacante suplantar una identidad después de que se haya comprometido una clave privada. Las CA pueden revocar un certificado comprometido, lo que impide su posterior uso. Los certificados firmados por una CA son, por lo tanto, más seguros para su uso en un entorno de producción, aunque los certificados autofirmados son más convenientes para un sistema de prueba.

Para obtener información completa sobre la creación y gestión de certificados, consulte "Trabajar con SSL/TLS en z/OS" en la página 322.

En esta colección de temas se presentan algunas de las tareas que forman parte de la configuración de las comunicaciones SSL o TLS, y se proporciona una guía paso a paso para completar estas tareas.

Es posible que también desee probar la autenticación de cliente SSL o TLS, que es una parte opcional de los protocolos. Durante el reconocimiento SSL o TLS, el cliente TLS o SSL siempre obtiene y valida un certificado digital del servidor. Con la implementación de IBM MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En z/OS, el cliente SSL o TLS envía un certificado solo si tiene uno de los siguientes certificados:

- Solo para un canal compartido, un certificado con una etiqueta en el formato `ibmWebSphereMQ` seguida del nombre del grupo de compartición de colas, por ejemplo `ibmWebSphereMQQSG1`
- Un certificado con una etiqueta en el formato `ibmWebSphereMQ` seguida del nombre del gestor de colas, por ejemplo `ibmWebSphereMQQM1`
- Un certificado predeterminado (que puede ser el certificado `ibmWebSphereMQ`).

Si el canal es compartido, el canal intenta primero encontrar un certificado para el grupo de compartición de colas. Si no encuentra un certificado para un grupo de compartición de colas, intenta encontrar un certificado para el gestor de colas.

En z/OS, IBM MQ utiliza el prefijo `ibmWebSphereMQ` en una etiqueta para evitar confusiones con certificados de otros productos.

El servidor SSL o TLS siempre valida el certificado de cliente si se envía uno. Si el cliente SSL o TLS no envía un certificado, la autenticación falla solo si el extremo del canal que actúa como servidor SSL o TLS se define con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información, consulte Conexión de dos gestores de colas utilizando SSL o TLS.

Trabajar con SSL/TLS

Estos temas proporcionan instrucciones para realizar tareas individuales relacionados con la utilización de TLS con IBM MQ.

Muchos de ellos se utilizan como pasos de las tareas de nivel superior que se describen en los apartados siguientes:

- "Identificación y autenticación de usuarios" en la página 334
- "Autorización del acceso a objetos" en la página 355
- "Confidencialidad de mensajes" en la página 426
- "Integridad de datos de mensajes" en la página 464
- "Mantenimiento de la seguridad de los clústeres" en la página 465

IBM i Trabajar con SSL/TLS en IBM i

Esta colección de temas ofrece instrucciones de tareas individuales relacionadas con TLS (Seguridad de la capa de transporte) en IBM MQ for IBM i.

Para IBM i, el soporte para TLS forma parte integral del sistema operativo. Asegúrese de que ha instalado los requisitos previos que se listan en Requisitos de hardware y software en IBM i.

En IBM i, las claves y los certificados digitales se gestionan con la herramienta Gestor de certificados digitales (DCM).

Acceso a DCM

Siga estas instrucciones para acceder a la interfaz del DCM.

Acerca de esta tarea

Realice los pasos siguientes en un navegador web que admita marcos.

Procedimiento

1. Vaya a `http://`*`machine`*`.domain:2001` o `https://`*`machine`*`.domain:2010`, donde *máquina* es el nombre del sistema.
2. Escriba un perfil de usuario y una contraseña válidos cuando se le solicite.

   Asegúrese de que el perfil de usuario tenga las autorizaciones especiales *ALLOBJ y *SECADM que le permitan crear almacenes de certificados nuevos. Si no tiene las autorizaciones especiales, sólo puede gestionar los certificados personales o ver las firmas de objetos correspondientes a los objetos para los que tiene autorización. Si tiene autorización para utilizar una aplicación de firma de objetos, también puede firmar objetos desde DCM.
3. En la página Configuraciones de Internet, pulse **Gestor de certificados digitales**.

   Se visualiza la página Gestor de certificados digitales.

Asignación de un certificado a un gestor de colas en IBM i

Utilice DCM para asignar un certificado a un gestor de consultas.

Utilice una gestión de certificados digitales de IBM i tradicional para asignar un certificado a un gestor de colas. Esto significa que puede especificar que un gestor de colas utilice el almacén de certificados del sistema, y que el gestor de colas se registre para su uso como aplicación con el Gestor de certificados digitales (DCM). Para ello, cambie el valor del atributo **SSLKEYR** del gestor de colas a *SYSTEM.

Cuando el parámetro **SSLKEYR** se cambia a *SYSTEM, IBM MQ registra el gestor de colas como una aplicación de servidor con una etiqueta de aplicación exclusiva de QIBM_WEBSPHERE_MQ_QMGRNAME y una etiqueta con una descripción de Qmgrname (WMQ). Tenga en cuenta que los atributos **CERTLABL** del canal no se utilizan si utiliza el almacén de certificados *SYSTEM. El gestor de colas aparece entonces como una aplicación de servidor en el Gestor de certificados digitales, y puede asignar a esta aplicación cualquier certificado de servidor o de cliente en el almacén de certificados.

Como el gestor de colas está registrado como una aplicación, se pueden efectuar las funciones avanzadas de DCM, como la definición de listas de CA fiables.

Si el parámetro **SSLKEYR** se cambia a un valor distinto de *SYSTEM, IBM MQ anula el registro del gestor de colas como una aplicación con el Certificate Managerdigital. Si se suprime un gestor de colas, también se anula el registro del DCM. Un usuario con autorización *SECADM suficiente también puede añadir o eliminar manualmente aplicaciones de DCM.

Configuración de un repositorio de claves en IBM i

Se debe configurar un depósito de claves en ambos extremos de la conexión. Se pueden utilizar los almacenes de certificados predeterminados o puede crear el suyo propio.

Una conexión TLS requiere un *depósito de claves* en cada extremo de la conexión. Cada gestor de colas y IBM MQ MQI client debe tener acceso a un repositorio de claves. Si desea acceder al repositorio de claves utilizando un nombre de archivo y contraseña (es decir, sin utilizar la opción *SYSTEM), asegúrese de que el perfil de usuario QMQM tiene las autorizaciones siguientes:

- Autorización de ejecución para el directorio que contiene el depósito de claves
- Autorización de lectura para el archivo que contiene el depósito de claves

Consulte “Repositorio de claves SSL/TLS” en la página 25 para obtener más información. Tenga en cuenta que los atributos de canal **CERTLABL** no se utilizan si utiliza el almacén de certificados *SYSTEM.

En IBM i, los certificados digitales se almacenan en un almacén de certificados que se gestiona con DCM. Estos certificados digitales tienen etiquetas, que asocian el certificado con un gestor de colas o un IBM MQ MQI client. TLS utiliza los certificados con fines de autenticación.

La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.

El nombre del almacén de certificados del IBM MQ MQI client o gestor de colas consta de una vía de acceso y un nombre de raíz. La vía de acceso predeterminada es `/QIBM/UserData/ICSS/Cert/Server/` y el nombre de raíz predeterminado es `Default`. En IBM i, el almacén de certificados predeterminado, `/QIBM/UserData/ICSS/Cert/Server/Default.kdb`, también se conoce como `*SYSTEM`. Opcionalmente, puede definir su propia vía de acceso y nombre de raíz.

Si define su propia vía de acceso o nombre de archivo, establezca los permisos del archivo para controlar estrechamente el acceso al mismo.

En el apartado “Cambiar la ubicación del repositorio de claves para un gestor de colas en IBM i” en la página 280 se indica cómo especificar el nombre de almacén de certificados. Puede especificar el nombre del almacén de certificados antes o después de crear el almacén de certificados.

Nota: Las operaciones que puede realizar con DCM pueden estar limitadas por la autorización de su perfil de usuario. Por ejemplo, necesita las autorizaciones *ALLOBJ y *SECADM para crear un certificado de CA.

Crear un almacén de certificados en IBM i
Si no desea utilizar el almacén de certificados predeterminado, siga este procedimiento para crear el suyo propio.

Acerca de esta tarea

Cree un almacén de certificados nuevo sólo si no desea utilizar el almacén de certificados predeterminado de IBM i.

Para especificar que se va a utilizar el almacén de certificados del sistema IBM i , cambie el valor del atributo SSLKEYR del gestor de colas a *SYSTEM. Este valor indica que el gestor de colas utiliza el almacén de certificados del sistema, y el gestor de colas se registra para su uso como aplicación con el Gestor de certificados digitales (DCM).

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en “Acceso a DCM” en la página 278
2. En el panel de navegación, pulse **Crear almacén de certificados nuevo**.
   La página Crear almacén de certificados nuevo se visualiza en la sección de tareas.
3. En la sección de tareas, seleccione **Otro almacén de certificados del sistema** y pulse **Continuar**.
   Se visualiza la página Crear un certificado en almacén de certificados nuevo en la sección de tareas.
4. Seleccione **No - No crear un certificado en el almacén de certificados** y pulse **Continuar**.
   La página Nombre y contraseña de almacén de certificados se visualiza en la sección de tareas.
5. En el campo **Vía de acceso de almacén de certificados y nombre de archivo** , escriba una vía de acceso y un nombre de archivo de IFS, por ejemplo `/QIBM/UserData/mqm/qmgrs/qm1/key.kdb`
6. Escriba una contraseña en el campo **Contraseña** y vuélvala a escribir en el campo **Confirmar contraseña**. Pulse **Continuar**.
   Anote la contraseña (que es sensible a mayúsculas y minúsculas) porque la necesitará cuando oculte la clave del repositorio.
7. Para salir del DCM, cierre la ventana del navegador.

Qué hacer a continuación

Cuando haya creado el almacén de certificados mediante DCM, asegúrese de ocultar la contraseña, tal como se describe en “Ocultación de la contraseña del almacén de certificados en sistemas IBM i” en la página 280

Tareas relacionadas

“Importar un certificado a un repositorio de claves en IBM i” en la página 285

Siga este procedimiento para importar un certificado.

Ocultación de la contraseña del almacén de certificados en sistemas IBM i

Oculte la contraseña del almacén de certificados utilizando mandatos CL.

Las siguientes instrucciones se aplican a la ocultación de la contraseña del almacén de certificados en IBM i para un gestor de colas. De forma alternativa, para un IBM MQ MQI client, si no está utilizando el almacén de certificados *SYSTEM (es decir, el entorno MQSSLKEYR está establecido en un valor distinto de *SYSTEM), siga el procedimiento descrito en la sección “Ocultar la contraseña del almacén de certificados” en la página 288 de “Programa de utilidad de Cliente SSL de IBM MQ (amqrsslc) para IBM i” en la página 287.

Si ha especificado que debe utilizarse el almacén de certificados *SYSTEM (cambiando el valor del atributo SSLKEYR del gestor de colas a *SYSTEM), no debe seguir estos pasos.

Cuando haya creado el almacén de certificados mediante DCM, utilice los mandatos siguientes para ocultar la contraseña:

```
STRMQM MQMNAME('queue_manager_name')
CHGMQM MQMNAME('queue_manager_name') SSLKEYRPWD('password')
```

La contraseña distingue entre mayúsculas y minúsculas. Debe especificarse entre comillas simples exactamente como lo ha hecho en el paso 6 de “Crear un almacén de certificados en IBM i” en la página 279.

Nota: Si no utiliza el almacén de certificados del sistema predeterminado, y no oculta la contraseña, los intentos de iniciar los canales TLS no se ejecutarán correctamente porque éstos no podrán obtener la contraseña necesaria para acceder al almacén de certificados.

Localizar el repositorio de claves para un gestor de colas en IBM i

Utilice este procedimiento para obtener la ubicación del almacén de certificados del gestor de colas.

Procedimiento

1. Visualice los atributos del gestor de colas, utilizando el siguiente mandato:

   ```
   DSPMQM MQMNAME('queue manager name')
   ```

2. Examine la salida del mandato para localizar la vía de acceso y nombre de raíz del almacén de certificados.

   Por ejemplo: `/QIBM/UserData/ICSS/Cert/Server/Default`, donde `/QIBM/UserData/ICSS/Cert/Server` es la vía de acceso y `Default` es el nombre de raíz.

Cambiar la ubicación del repositorio de claves para un gestor de colas en IBM i

Cambie la ubicación del almacén de certificados del gestor de colas utilizando CHGMQM o ALTER QMGR.

Procedimiento

Utilice el mandato CHGMQM o el mandato MQSC ALTER QMGR para establecer el atributo de repositorio de claves del gestor de colas.

a) Utilizando CHGMQM: `CHGMQM MQMNAME('qm1') SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')`

b) Utilizando ALTER QMGR: `ALTER QMGR SSLKEYR('/QIBM/UserData/ICSS/Cert/Server/MyKey')`

En ambos casos, el almacén de certificados tiene el nombre de archivo completo: `/QIBM/UserData/ICSS/Cert/Server/MyKey.kdb`

Qué hacer a continuación

Cuando cambia la ubicación del almacén de certificados de un gestor de colas, los certificados no se transfieren desde la ubicación antigua. Si los certificados de CA que se instalaron de antemano al crear el almacén de certificados son insuficientes, debe llenar el nuevo almacén de certificados con certificados, tal como se describe en "Importar un certificado a un repositorio de claves en IBM i" en la página 285. También debe ocultar la contraseña para la nueva ubicación, según se describe en "Ocultación de la contraseña del almacén de certificados en sistemas IBM i" en la página 280.

Crear una entidad emisora de certificados y un certificado para pruebas en IBM i

Utilice este procedimiento para crear un certificado de CA local para firmar peticiones de certificado, y para crear e instalar el certificado de CA.

Antes de empezar

Las instrucciones de este tema presuponen que no existe una autorización de certificado (CA) local. Si no existe una CA local, vaya a "Solicitar un certificado de servidor en IBM i" en la página 282.

Acerca de esta tarea

Los certificados de CA que se proporcionan cuando se instala TLS están firmados por la CA emisora. En IBM i, puede generar una entidad emisora de certificados local que pueda firmar certificados de servidor para probar las comunicaciones TLS en el sistema. Siga estos pasos en un navegador web para crear un certificado de CA local:

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
2. En el panel de navegación, pulse **Crear una Entidad emisora de certificados**.

   La página Crear una Entidad emisora de certificados se visualiza en la sección de tareas.
3. Escriba una contraseña en el campo **Contraseña del almacén de certificados** y vuelva a escribirla en el campo **Confirmar contraseña**.
4. Escriba un nombre en el campo **Nombre de entidad emisora de certificados (CA)** , por ejemplo TLS `Test Certificate Authority`.
5. Escriba valores adecuados en los campos **Nombre común** y **Organización** y seleccione un país. En cuanto a los campos opcionales restantes, escriba los valores que necesite.
6. Escriba un periodo de validez para la CA local en el campo **Periodo de validez**.

   El valor predeterminado es 1095 días.
7. Pulse **Continuar**.

   Se crea la CA y DCM crea un almacén de certificados y un certificado de CA para la CA local.
8. Pulse **Instalar certificado**.

   Se visualiza el recuadro de diálogo del gestor de descargas.
9. Escriba el nombre de vía de acceso completo del archivo temporal en el que desea almacenar el certificado de CA y pulse **Guardar**.
10. Cuando el proceso de descarga haya finalizado, pulse **Abrir**.

    Se visualiza la ventana Certificado.
11. Pulse **Instalar certificado**.

    Se visualiza el asistente Importar certificado.
12. Pulse **Siguiente**.

13. Seleccione **Seleccionar automáticamente el almacén de certificados basándose en el tipo de certificado** y pulse **Siguiente**.
14. Pulse **Finalizar**.

    Se visualiza una ventana de confirmación.
15. Pulse **Aceptar**.
16. En la ventana Certificado, pulse **Aceptar**.
17. Pulse **Continuar**.

    Se visualiza la página Política de Entidad emisora de certificados en la sección de tareas.
18. En el campo **Permitir la creación de certificados de usuario**, seleccione **Sí**.
19. En el campo **Periodo de validez**, escriba el periodo de validez de los certificados emitidos por la CA local.

    El valor predeterminado es 365 días.
20. Pulse **Continuar**.

    Se visualiza la página Crear un certificado en almacén de certificados nuevo en la sección de tareas.
21. Compruebe que ninguna de las aplicaciones está seleccionada.
22. Pulse **Continuar** para completar la configuración de la CA local.

Solicitar un certificado de servidor en IBM i

Los certificados digitales protegen contra la suplantación de identidad, certificando que una clave pública pertenece a una entidad especificada. Se puede solicitar un nuevo certificado de servidor de una entidad emisora de certificados utilizando el Gestor de certificados digitales (DCM).

Acerca de esta tarea

Realice los pasos siguientes en un navegador web:

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado “Acceso a DCM” en la página 278.
2. En el panel de navegación, pulse **Seleccionar un almacén de certificados**.

   La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
3. Seleccione el almacén de certificados que desea utilizar y pulse **Continuar**.
4. Opcional: Si ha seleccionado ***SYSTEM** en el paso 3, entre la contraseña del almacén del sistema y pulse **Continuar**.
5. Opcional: Si ha seleccionado **Otro almacén de certificados del sistema** en el paso 3, en el campo **Vía de acceso y nombre de archivo del almacén de certificados**, escriba la vía de acceso y nombre de archivo de IFS que estableció cuando creó el almacén de certificados. Además, escriba una contraseña en el campo **Contraseña del almacén de certificados**. A continuación, pulse **Continuar**
6. En el panel de navegación, pulse **Crear certificado**.
7. En la sección de tareas, seleccione el botón **Certificado de servidor o de cliente** y pulse **Continuar**.

   La página Seleccionar una Entidad emisora de certificados (CA) se visualiza en la sección de tareas.
8. Si tiene una CA local en la estación de trabajo, elija la CA local o una CA comercial para firmar el certificado. Seleccione el botón de selección correspondiente a la CA que desee y pulse **Continuar**.

   La página Crear un certificado se visualiza en la sección de tareas.
9. Opcional: Para un gestor de colas, en el campo **Etiqueta de certificado**, escriba la etiqueta del certificado.

   La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.

   Por ejemplo, para el gestor de colas QM1, escriba `ibmwebspheremqqm1` para utilizar el valor predeterminado.

10. Opcional: Para un IBM MQ MQI client, en el campo **Etiqueta de certificado**, escriba `ibmwebspheremq` seguido del ID de usuario de inicio de sesión en minúsculas. Por ejemplo, escriba `ibmwebspheremqmyuserID`
11. Escriba valores adecuados en los campos **Nombre común** y **Organización** y seleccione un país. En cuanto a los campos opcionales restantes, escriba los valores que necesite.

Resultados

Si ha seleccionado una CA comercial para firmar el certificado, DCM crea una solicitud de certificado en formato PEM (Privacy-Enhanced Mail). Reenvíe la solicitud a la CA que haya elegido.

Si ha seleccionado la CA local para firmar el certificado, DCM le informa de que el certificado se ha creado en el almacén de certificados y que se puede utilizar.

Solicitar un certificado de servidor para IBM Key Manager en IBM i

Siga este procedimiento para crear un certificado firmado por la entidad emisora de certificados (CA) local, o para solicitar un certificado de servidor firmado por una CA comercial para importarlo al programa de utilidad IBM Key Management (iKeyman).

Acerca de esta tarea

Debe utilizarse un certificado de usuario cuando el Gestor de certificados digitales (DCM) actúa como gestor de certificados para IBM MQ en varias plataformas. Para certificados personales distribuidos a otras plataformas y para importarlos al programa de utilidad iKeyman, realice los pasos siguientes en un navegador web:

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
2. En el panel de **navegación**, pulse **Crear certificado**.

   La página **Crear certificado** se visualiza en la sección de tareas.
3. En el panel **Crear certificado**, seleccione el botón **Certificado de usuario** y pulse **Continuar**.

   Se visualiza la página **Crear certificado de usuario**.
4. En el panel **Crear certificado de usuario**, rellene los campos obligatorios bajo Información del certificado para **Nombre de organización**, **Estado** o **provincia**, **País** o **región**. Opcionalmente, entre valores en los campos **Unidad de organización** y **Localidad** o **ciudad**. Pulse **Continuar**.

   El **Nombre común** se establece automáticamente en el ID de usuario con el que ha iniciado la sesión en el sistema iSeries.
5. En el siguiente panel **Crear certificado de usuario**, pulse **Instalar certificado** y pulse **Continuar**.

   Se visualiza un mensaje que indica que `Se ha instalado su certificado personal. Debería guardar una copia de seguridad de este certificado.`
6. Pulse **Aceptar**.
7. Dependiendo del navegador web que haya utilizado para acceder a DCM, realice los pasos siguientes:
   a) Para Microsoft Edge, elija: **Herramientas > Opciones de Internet > separador Contenido > botón Certificados > separador Personal >**. Seleccione el certificado y pulse **Exportar**.
   b) Para Mozilla Firefox, seleccione: **Herramientas > Opciones > Avanzado> Pestaña Cifrado > botón Ver certificados > pestaña Sus certificados >**. Seleccione el certificado y pulse **Copia de seguridad**. Seleccione la vía de acceso y el nombre de archivo y pulse **Aceptar**.
8. Transfiera el certificado exportado al sistema remoto utilizando FTP en formato binario.
9. Añada el certificado exportado en el paso 7 al programa de utilidad iKeyman en la base de datos de claves.
   a) Si el certificado se ha guardado utilizando Microsoft Edge, utilice las instrucciones que se describen en Importación desde un archivo Microsoft `.pfx`.
   b) Si el certificado se ha guardado utilizando Mozilla Firefox, siga las instrucciones que se indican en Importación de un certificado personal a un repositorio de claves.

Durante la importación, asegúrese de que el nombre de etiqueta del certificado personal y el certificado de firmante se cambien de modo que quede reflejado lo que espera IBM MQ. La etiqueta es el valor del atributo IBM MQ **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.

Añadir certificados de servidor a un repositorio de claves en IBM i

Siga este procedimiento para añadir un certificado solicitado al repositorio de claves.

Acerca de esta tarea

Después de que la CA envíe un nuevo certificado de servidor, debe añadirlo al almacén de certificados desde el que ha generado la solicitud. Si la CA envía el certificado como parte de un mensaje de correo electrónico, copie el certificado en un archivo aparte.

Nota:

- No necesita efectuar este procedimiento si la CA local firma el certificado de servidor.
- Antes de importar un certificado de servidor con el formato PKCS #12 a DCM, deberá importar primero el certificado de CA correspondiente.

Utilice el siguiente procedimiento para recibir un certificado de servidor en el almacén de certificados del gestor de colas:

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado “Acceso a DCM” en la página 278.
2. En la categoría de tareas **Gestionar certificados** del panel de navegación, pulse **Importar certificado**.

   Se visualiza la página Importar certificado en la sección de tareas.
3. Seleccione el botón de selección correspondiente al tipo de certificado y pulse **Continuar**.

   Se visualiza la página Importar certificado de servidor o de cliente o Importar certificado de Entidad emisora de certificados (CA) en la sección de tareas.
4. En el campo **Importar archivo**, escriba el nombre de archivo del certificado que desea importar y pulse **Continuar**.

   DCM determina automáticamente el formato del archivo.
5. Si el certificado es un certificado de **Servidor o cliente**, escriba la contraseña en la sección de tareas y pulse **Continuar**.

   DCM le informa de que se ha importado el certificado.

Exportar un certificado desde un repositorio de claves en IBM i

Exportar un certificado exporta ambas claves, la pública y privada. Esta acción se deberá realizar con extrema cautela, ya que pasar una clave privada comprometería por completo la seguridad.

Antes de empezar

Cuando se comparte un certificado de usuario con otro usuario, se intercambian claves públicas. Este proceso se describe en la tarea 5 de **. Compartir certificados** en la Guía de inicio rápido para AMS en UNIX. Al exportar un certificado tal como se describe aquí, se exportan ambas claves, pública y privada. Esta acción se deberá realizar con extrema cautela, ya que pasar una clave privada comprometería por completo la seguridad.

Acerca de esta tarea

Realice los pasos siguientes en el sistema desde el que desea exportar el certificado:

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
2. En el panel de navegación, pulse **Seleccionar un almacén de certificados**.

   La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
3. Seleccione el almacén de certificados que desea utilizar y pulse **Continuar**.
4. Opcional: Si ha seleccionado ***SYSTEM** en el paso 3, entre la contraseña del almacén del sistema y pulse **Continuar**.
5. Opcional: Si ha seleccionado **Otro almacén de certificados del sistema** en el paso 3, en el campo **Vía de acceso y nombre de archivo del almacén de certificados**, escriba la vía de acceso y nombre de archivo de IFS que estableció cuando creó el almacén de certificados y escriba una contraseña en el campo **Contraseña del almacén de certificados**. A continuación, pulse **Continuar**
6. En la categoría de tareas **Gestionar certificados** del panel de navegación, pulse **Exportar certificado**.

   La página Exportar un certificado se visualiza en la sección de tareas.
7. Seleccione el botón de selección correspondiente al tipo de certificado y pulse **Continuar**.

   Se visualiza la página Exportar certificado de servidor o de cliente o Exportar certificado de Entidad emisora de certificados (CA) en la sección de tareas.
8. Seleccione el certificado que desea exportar.
9. Seleccione el botón de selección para especificar si desea exportar el certificado a un archivo o directamente a otro almacén de certificados.
10. Si ha seleccionado exportar un certificado de servidor o de cliente a un archivo, facilite la siguiente información:
    - La vía de acceso y nombre de archivo de la ubicación donde desea almacenar el certificado exportado.
    - Para un certificado personal, la contraseña que se utiliza para cifrar el certificado exportado y el release de destino. En el caso de certificados de CA, no necesita especificar la contraseña.
11. Si ha seleccionado exportar un certificado directamente a otro almacén de certificados, especifique el almacén de certificados de destino y la contraseña.
12. Pulse **Continuar**.

Importar un certificado a un repositorio de claves en IBM i

Siga este procedimiento para importar un certificado.

Antes de empezar

Antes de importar un certificado personal con el formato PKCS #12 a DCM, deberá importar primero el certificado de CA correspondiente.

Acerca de esta tarea

Realice estos pasos en la máquina a la que desea importar el certificado.

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
2. En el panel de navegación, pulse **Seleccionar un almacén de certificados**.

   La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
3. Seleccione el almacén de certificados que desea utilizar y pulse **Continuar**.
4. Opcional: Si ha seleccionado ***SYSTEM** en el paso 3, entre la contraseña del almacén del sistema y pulse **Continuar**.
5. Opcional: Si ha seleccionado **Otro almacén de certificados del sistema** en el paso 3, en el campo **Vía de acceso y nombre de archivo del almacén de certificados**, escriba la vía de acceso y nombre de

archivo de IFS que estableció cuando creó el almacén de certificados y escriba una contraseña en el campo **Contraseña del almacén de certificados**. A continuación, pulse **Continuar**

6. En la categoría de tareas **Gestionar certificados** del panel de navegación, pulse **Importar certificado**.
   La página Importar certificado se visualiza en la sección de tareas.
7. Seleccione el botón de selección correspondiente al tipo de certificado y pulse **Continuar**.
   Se visualiza la página Importar certificado de servidor o de cliente o Importar certificado de Entidad emisora de certificados (CA) en la sección de tareas.
8. En el campo **Importar archivo**, escriba el nombre de archivo del certificado que desea importar y pulse **Continuar**.
   DCM determina automáticamente el formato del archivo.
9. Si el certificado es un certificado de **Servidor o cliente**, escriba la contraseña en la sección de tareas y pulse **Continuar**. DCM le informa de que se ha importado el certificado.

Suprimir certificados en IBM i

Utilice este procedimiento para suprimir certificados personales.

Procedimiento

1. Acceda a la interfaz DCM, tal y como se describe en el apartado “Acceso a DCM” en la página 278.
2. En el panel de navegación, pulse **Seleccionar un almacén de certificados**.
   La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
3. Seleccione el recuadro **Otro almacén de certificados del sistema** y pulse **Continuar**.
   Se visualiza la página Almacén de certificados y contraseña.
4. En el campo **Vía de acceso y nombre de archivo del almacén de certificados**, escriba la vía de acceso y nombre de archivo de IFS que estableció cuando creó el almacén de certificados.
5. Escriba una contraseña en el campo **Contraseña del almacén de certificados**. Pulse **Continuar**.
   La página Almacén de certificados actual se visualiza en la sección de tareas.
6. En la categoría de tareas **Gestionar certificados** del panel de navegación, pulse **Suprimir certificado**.
   La página Confirmar supresión de certificado se visualiza en la sección de tareas.
7. Seleccione el certificado que desea suprimir. Pulse **Suprimir**.
8. Pulse **Sí** para confirmar que desea suprimir el certificado. De lo contrario, pulse **No**.
   DCM le informa si ha suprimido el certificado.

*Utilización del almacén de certificados *SYSTEM para la autenticación unidireccional en IBM i*

Siga estas instrucciones para configurar la autenticación unidireccional.

Antes de empezar

- Cree un gestor de colas, canales y colas de transmisión.
- Cree un certificado de servidor o de cliente en el gestor de colas del servidor.
- Transfiera el certificado de CA al gestor de colas del cliente e impórtelo al repositorio de claves.
- Inicie un escucha en los gestores de colas del servidor y del cliente.

Acerca de esta tarea

Para utilizar la autenticación unidireccional, utilizando un sistema que ejecuta IBM i como servidor TLS, establezca el parámetro Repositorio de claves SSL (SSLKEYR) en *SYSTEM. Este valor registra el gestor de colas de IBM MQ como una aplicación. Podrá entonces asignar un certificado al gestor de colas para permitir la autenticación unidireccional.

También puede utilizar almacenes de claves privados para implementar la autenticación unidireccional creando un certificado ficticio para el gestor de colas del cliente en el depósito de claves.

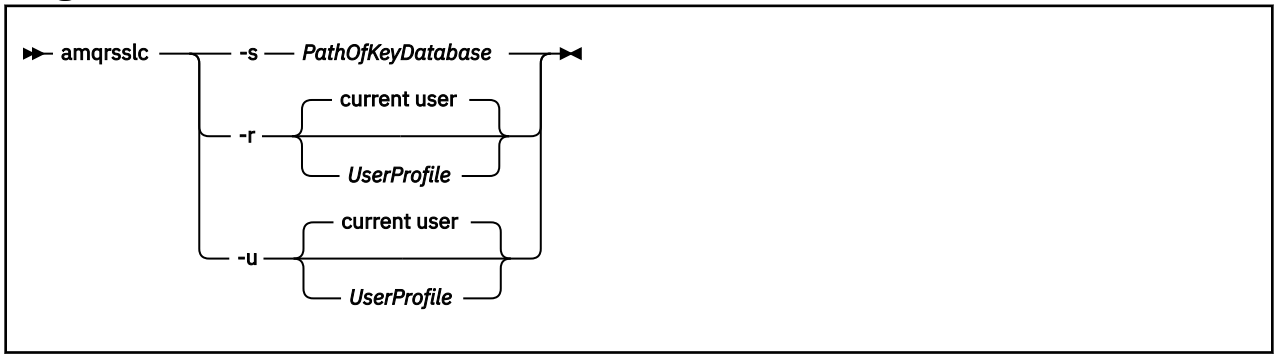
Procedimiento

1. Realice los pasos siguientes en los gestores de colas del servidor y del cliente:
   a) Modifique el gestor de colas para establecer el parámetro SSLKEYR, emitiendo el mandato `CHGMQM MQMNAME(SSL) SSLKEYR(*SYSTEM)`.
   b) Oculte la contraseña para el depósito de claves predeterminado, emitiendo el mandato `CHGMQM MQMNAME(SSL) SSLKEYRPWD('xxxxxxx')`.
      La contraseña debe estar entre comillas simples.
   c) Modifique los canales para que tengan la CipherSpec correcta en el parámetro SSLCIPHER.
   d) Renueve la seguridad TLS emitiendo el mandato `RFRMQMAUT QMNAME(QMGRNAME) TYPE(*SSL)`.
2. Asigne el certificado al gestor de colas del servidor utilizando DCM, como se indica a continuación:
   a) Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
   b) En el panel de navegación, pulse **Seleccionar un almacén de certificados**.
      La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
   c) Seleccione el almacén de certificados *SYSTEM y pulse **Continuar**.
   d) En el panel de la izquierda, expanda **Gestionar aplicaciones**.
   e) Seleccione **Ver definición de aplicación** para comprobar que el gestor de colas se ha registrado como una aplicación.
      `SSL (WMQ)` aparece listado en la tabla.
   f) Seleccione **Actualizar asignación de certificados**.
   g) Seleccione **Servidor** y pulse **Continuar**.
   h) Seleccione QMGRNAME (WMQ) y pulse **Actualizar asignación de certificados**.
   i) Seleccione el certificado y pulse **Asignar nuevo certificado**. Se abre una ventana que le informa de que el certificado se ha asignado a la aplicación.

Programa de utilidad de Cliente SSL de IBM MQ (amqrsslc) para IBM i

El programa de utilidad de Cliente SSL de IBM MQ (amqrsslc) para IBM i lo utiliza el IBM MQ MQI client En los sistemas IBM i para registrar o anular el registro del perfil de usuario del cliente o para ocultar la contraseña del almacén de certificados. El programa de utilidad sólo lo puede ejecutar un usuario con un perfil con autorización especial *ALLOBJ o un miembro del grupo QMQMADM que tenga opciones para crear o suprimir registros de aplicación en el Gestor de certificados digitales (DCM).

Diagrama de sintaxis



Registrar el perfil de usuario del cliente

Si IBM MQ MQI client utiliza el almacén de certificados *SYSTEM, debe registrar el perfil de usuario de cliente (usuario de inicio de sesión) para utilizarlo como una aplicación con Digital Certificate Manager (DCM).

Si desea registrar el perfil de usuario del cliente, ejecute el programa **amqrsslc** con la opción `-r` con *PerfilUsuario*. El perfil de usuario utilizado al llamar a **amqrsslc** debe tener autorización *USE. Al especificar el *PerfilUsuario* con la opción `-r`, se registra el *PerfilUsuario* como una aplicación de servidor con una etiqueta de aplicación exclusiva de QIBM_WEBSPHERE_MQ_*PerfilUsuario* y una etiqueta con una descripción de *PerfilUsuario* (WMQ). Esta aplicación de servidor se mostrará entonces en el DCM y podrá asignar a esta aplicación cualquier certificado de servidor o de cliente del almacén del sistema.

Nota: Si un perfil de usuario no se especifica con la opción `-r`, se registra el perfil de usuario del usuario que ejecuta la herramienta **amqrsslc**.

El código siguiente utiliza **amqrsslc** para registrar un perfil de usuario. En el primer ejemplo, se registra el perfil de usuario especificado; en el segundo, es el perfil del usuario que ha iniciado la sesión:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-r' UserProfile)
CALL PGM(QMQM/AMQRSSLC) PARM('-r')
```

Anular el registro del perfil de usuario del cliente

Para anular el registro del perfil del cliente, ejecute el programa **amqrsslc** con la opción `-u` con *PerfilUsuario*. El perfil de usuario utilizado al llamar a **amqrsslc** debe tener autorización *USE. Al proporcionar al *PerfilUsuario* la etiqueta `-u` se anula el registro de *PerfilUsuario* con la etiqueta QIBM_WEBSPHERE_MQ_*PerfilUsuario* del DCM.

Nota: Si un perfil de usuario no se especifica con la opción –u, se anula el registro del perfil de usuario del usuario que ejecuta la herramienta **amqrsslc**.

El código siguiente utiliza **amqrsslc** para anular el registro de un perfil de usuario. En el primer ejemplo, se anula el registro del perfil de usuario especificado; en el segundo, es el perfil del usuario que ha iniciado la sesión:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-u' UserProfile)
CALL PGM(QMQM/AMQRSSLC) PARM('-u')
```

Ocultar la contraseña del almacén de certificados

Si el IBM MQ MQI client no utiliza el almacén de certificados *SYSTEM y utiliza otro almacén de certificados (es decir, MQSSLKEYR se establece en un valor distinto de *SYSTEM), la contraseña de la base de datos de claves debe estar oculta. Utilice la opción `-s` para ocultar la contraseña de la base de datos de claves.

En el código siguiente, el nombre de archivo completo del almacén de certificados es `/Path/Of/KeyDatabase/MyKey.kdb`:

```
CALL PGM(QMQM/AMQRSSLC) PARM('-s' '/Path/Of/KeyDatabase/MyKey')
```

La ejecución de este código da como resultado una solicitud de la contraseña de esta base de datos de claves. Esta contraseña está oculta en un archivo con el mismo nombre que la base de datos de claves con una extensión .kdb. Este archivo se almacena en la misma vía de acceso que la base de datos de claves. El ejemplo de código genera un archivo de ocultación `/Path/Of/KeyDatabase/MyKey.sth`. QMQM es el propietario de usuario y QMQMADM el propietario del grupo para este archivo. QMQM y QMQMADM tienen permiso de lectura y grabación, y otros perfiles sólo tienen permiso de lectura.

Cuándo entran en vigor los cambios en los certificados o en el almacén de certificados en IBM i

Cuando cambia los certificados de un almacén de certificados, o la ubicación del almacén de certificados, los cambios entran en vigor dependiendo del tipo de canal y de cómo se ejecuta el canal.

Los cambios efectuados en los certificados del almacén de certificados y en el atributo de repositorio de claves entran en vigor en las siguientes situaciones:

- Cuando un nuevo proceso de canal de salida individual ejecuta por primera vez un canal TLS.
- Cuando un nuevo proceso de canal de entrada individual TCP/IP recibe por primera vez una solicitud para iniciar un canal TLS.
- Cuando se emite el mandato MQSC REFRESH SECURITY TYPE(SSL) para renovar el entorno TLS de IBM MQ.
- Para los procesos de la aplicación cliente, cuando se cierra la última conexión TLS del proceso. La siguiente conexión TLS captará los cambios del certificado.
- Para canales que se ejecutan como hebras de un proceso de agrupación de procesos (amqrmppa), cuando se inicia o se reinicia el proceso de agrupación de procesos y ejecuta por primera vez un canal TLS. Si el proceso de agrupación de procesos ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).
- Para canales que se ejecutan como hebras del iniciador de canal, cuando se inicia o se reinicia el iniciador de canal y ejecuta por primera vez un canal TLS. Si el proceso iniciador de canal ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).
- Para canales que se ejecutan como hebras de un escucha TCP/IP, cuando se inicia o se reinicia el escucha y recibe por primera vez una solicitud para iniciar un canal TLS. Si el escucha ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).

Configurar el hardware de cifrado en IBM i

Utilice este procedimiento para configurar el Coprocesador criptográfico en IBM i

Antes de empezar

Asegúrese de que el perfil de usuario tenga las autorizaciones especiales *ALLOBJ y *SECADM para configurar el hardware del coprocesador.

Procedimiento

1. Vaya a `http://`*`machine`*`.domain:2001` o `https://`*`machine`*`.domain:2010`, donde *máquina* es el nombre del sistema.

   Se visualiza un recuadro de diálogo que le solicita un nombre de usuario y una contraseña.
2. Escriba un perfil de usuario y una contraseña válidos de IBM i.
3. Vaya a Criptografía y siga los correspondientes enlaces para obtener más información.

Qué hacer a continuación

Para obtener información más concreta sobre la configuración del Coprocesador criptográfico 4767, consulte Coprocesador criptográfico 4767.

ULW Trabajar con SSL/TLS en UNIX, Linux, and Windows

En sistemas UNIX, Linux, and Windows, el soporte TLS (Transport Layer Security) se instala con IBM MQ.

Para obtener más información sobre las políticas de validación de certificados, consulte Validación de certificados y diseño de políticas de confianza.

ULW *Utilización de runmqckm, runmqakm y strmqikm para gestionar certificados digitales*

En sistemas UNIX, Linux, and Windows, gestione las claves y los certificados digitales con la interfaz gráfica de usuario de **strmqikm** (iKeyman) o desde la línea de mandatos utilizando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

V 9.1.0

Atención: Los mandatos **runmqckm** y **strmqikm** se basan en Java Runtime Environment (JRE) de IBM MQ. Desde IBM MQ 9.1, si el JRE no está instalado, recibirá el mensaje AMQ9183.

- Para sistemas **UNIX and Linux** :
  - Utilice el mandato **strmqikm** (iKeyman) para iniciar la GUI de iKeyman.
  - Utilice el mandato **runmqckm** (iKeycmd) para realizar tareas con la interfaz de línea de mandatos de iKeycmd.
  - Utilice el mandato **runmqakm** (GSKCapiCmd) para realizar tareas con la interfaz de línea de mandatos runmqakm. La sintaxis de mandato para **runmqakm** es la misma que para **runmqckm**.

    Si necesita gestionar certificados TLS de modo que sean compatibles con FIPS, utilice el mandato **runmqakm** en lugar de los mandatos **runmqckm** o **strmqikm**.

  Consulte Gestión de claves y certificados para obtener una descripción detallada de las interfaces de línea de mandatos para los mandatos **runmqckm** y **runmqakm**.

  Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que iKeycmd e iKeyman son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas de Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas iKeyman e iKeycmd son de 32 bits en dichas plataformas.

  Para obtener más información, consulte GSKit: PKCS#11 y modalidad de direccionamiento de JRE IBM MQ.

  Antes de ejecutar el mandato **strmqikm** para iniciar la GUI de iKeyman, asegúrese de que trabaja en una máquina que puede ejecutar X Window System y haga lo siguiente:

  - Establezca la variable de entorno DISPLAY, por ejemplo:

    ```
    export DISPLAY=mypc:0
    ```

  - Asegúrese de que la variable de entorno PATH contiene **/usr/bin** y **/bin**. También es necesario para los mandatos **runmqckm** y**runmqakm**. Por ejemplo:

    ```
    export PATH=$PATH:/usr/bin:/bin
    ```

- Para sistemas **Windows** :
  - Utilice el mandato **strmqikm** para iniciar la GUI de iKeyman.
  - Utilice el mandato **runmqckm** para realizar tareas con la interfaz de línea de mandatos de iKeycmd.

    Si necesita gestionar certificados TLS de modo que sean compatibles con FIPS, utilice el mandato **runmqakm** en lugar de los mandatos **runmqckm** o **strmqikm**.
  - Utilice el mandato **runmqakm -keydb** con la opción *stashpw* o *stash*.

    Cuando se utiliza el mandato **runmqakm -keydb** de esta forma, por ejemplo:

    ```
    runmqakm -keydb -create -db key.kdb -pw secretpwd -stash
    ```

    el archivo resultante `.sth` no tiene permiso de lectura habilitado para el grupo `mqm`.

Sólo el creador puede leer el archivo. Tras crear un archivo stash utilizando el mandato **runmqakm**, compruebe los permisos de archivo y otorgue permiso a la cuenta de servicio que ejecuta el gestor de colas o a un grupo como mqm local.

Para solicitar el rastreo de TLS en sistemas UNIX, Linux o Windows, consulte strmqtrc.

Referencia relacionada

Mandatos runmqckm y runmqakm
En esta sección se describen los mandatos runmqckm y runmqakm según el objeto del mandato.

ULW *Configuración de un repositorio de claves en UNIX, Linux, and Windows*

Puede definir un repositorio de claves utilizando la interfaz gráfica de usuario de **strmqikm** (iKeyman), o desde la línea de mandatos mediante los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

Acerca de esta tarea

Una conexión TLS requiere un *depósito de claves* en cada extremo de la conexión. Cada gestor de colas de IBM MQ y IBM MQ MQI client debe tener acceso a un repositorio de claves. Para obtener más información, consulte “Repositorio de claves SSL/TLS” en la página 25.

En los sistemas UNIX, Linux, and Windows , los certificados digitales se almacenan en un archivo de base de datos de claves que se gestiona utilizando la interfaz de usuario de **strmqikm** , o utilizando los mandatos **runmqckm** o **runmqakm** . Estos certificados digitales tienen etiquetas. Una etiqueta específica asocia un certificado personal a un gestor de colas o un IBM MQ MQI client. TLS utiliza este certificado con fines de autenticación. En los sistemas UNIX, Linux, and Windows, IBM MQ utiliza el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado ibmwebspheremq con el nombre del gestor de colas o el ID de inicio de sesión de usuario de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.

El nombre de archivo de base de datos de claves consta de una vía de acceso y un nombre de raíz:

- En sistemas UNIX and Linux , la vía de acceso predeterminada para un gestor de colas (establecida al crear el gestor de colas) es /var/mqm/qmgrs/*queue_manager_name*/ssl.

  En sistemas Windows , la vía de acceso predeterminada es *MQ_INSTALLATION_PATH*\Qmgrs*queue_manager_name*\ssl, donde *MQ_INSTALLATION_PATH* es el directorio en el que está instalado IBM MQ . Por ejemplo, C:\Archivos de programa\IBM\MQ\Qmgrs\QM1\ssl.

  El nombre de raíz predeterminado es key. Opcionalmente, puede seleccionar su propia vía de acceso y nombre de raíz, pero la extensión debe ser .kdb.

  Si elige su propia vía de acceso o nombre de archivo, establezca los permisos del archivo para controlar estrechamente el acceso al mismo.
- Para un cliente IBM MQ, no hay ninguna vía de acceso ni nombre de raíz predeterminados. Controle estrechamente el acceso a este archivo. La extensión debe ser .kdb.

No cree repositorios de claves en un sistema de archivos sin soporte para bloqueos a nivel de archivo, por ejemplo, NFS versión 2 en los sistemas Linux.

Para obtener información sobre cómo comprobar y especificar el nombre de archivo de base de datos de claves, consulte “Cambiar la ubicación del repositorio de claves para un gestor de colas en UNIX, Linux, and Windows” en la página 296. Puede especificar el nombre de archivo de base de datos de claves antes o después de crear el archivo de base de datos de clave.

El ID de usuario desde el que ejecuta los mandatos **strmqikm** o **runmqckm** debe tener permiso de escritura para el directorio en el que se crea o actualiza el archivo de base de datos de claves. Para un gestor de colas que utiliza el directorio ssl predeterminado, el ID de usuario desde el que ejecuta **strmqikm** o **runmqckm** debe ser miembro del grupo mqm. Para un IBM MQ MQI client, si ejecuta **strmqikm** o **runmqckm** desde un ID de usuario distinto del que se ejecuta el cliente, debe modificar los permisos de archivo para permitir que el IBM MQ MQI client acceda al archivo de base de datos de claves en tiempo de ejecución. Para obtener más información, consulte “Acceso y protección de los archivos de

base de datos de claves en Windows" en la página 293 o "Acceso y protección de los archivos de base de datos de claves en sistemas UNIX and Linux" en la página 294.

En **strmqikm** o **runmqckm** para IBM WebSphere MQ 7.0, las nuevas bases de datos de claves se rellenan automáticamente con un conjunto de certificados de entidad emisora de certificados (CA) predefinidos. En **strmqikm** o **runmqckm** para IBM MQ 8.0, las bases de datos de claves no se llenan automáticamente, lo que hace que la configuración inicial sea más segura porque sólo incluye los certificados de CA que desea, en el archivo de base de datos de claves.

Nota: Debido a este cambio en el comportamiento de GSKit 8.0 que hace que los certificados de CA ya no se añadan automáticamente al repositorio, debe añadir manualmente los certificados de CA preferidos. Este cambio de comportamiento le proporciona más control granular sobre los certificados de CA utilizados. Consulte "Adición de certificados de CA predeterminados a un repositorio de claves vacío en UNIX, Linux, and Windows con GSKit 8.0" en la página 294.

Puede crear la base de datos de claves utilizando la línea de mandatos o utilizando la interfaz de usuario de **strmqikm** (iKeyman).

Nota: Si tiene que gestionar los certificados TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**. La interfaz de usuario de **strmqikm** no proporciona una opción compatible con FIPS.

Procedimiento

Cree una base de datos de claves utilizando la línea de mandatos.

1. Ejecute uno de los mandatos siguientes:
   - Mediante **runmqckm**:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Mediante **runmqakm**:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
     -stash -fips -strong
     ```

   donde:

   -db *nombrearchivo*
   Especifica el nombre completo de una base de datos de claves CMS y debe tener una extensión de archivo de `.kdb`.

   -pw *contraseña*
   Especifica la contraseña para la base de datos de claves CMS.

   -type *cms*
   Especifica el tipo de base de datos. (Para IBM MQ, debe ser `cms`.)

   -stash
   Guarda la contraseña de la base de datos de claves en un archivo.

   -fips
   Especifica que el mandato se ejecuta en modalidad FIPS. Cuando está en modalidad FIPS, el componente ICC utiliza los algoritmos que se han validado con FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

   -strong
   Comprueba que la contraseña especificada cumple los requisitos mínimos de validez de contraseña. Los requisitos mínimos para una contraseña son los siguientes:

   - La contraseña debe tener una longitud mínima de 14 caracteres.
   - La contraseña debe contener un mínimo de un carácter en minúsculas, un carácter en mayúsculas, y un dígito o un carácter especial. Los caracteres especiales incluyen el asterisco

(*), el signo de dólar ($), el signo de número (#) y el signo de porcentaje (%). Un espacio se clasifica como un carácter especial.

- Cada carácter puede aparecer un máximo de tres veces en una contraseña.
- Dos es el número máximo de caracteres consecutivos que pueden ser idénticos.
- Todos los caracteres pertenecen al juego de caracteres ASCII imprimibles estándar dentro del rango entre 0x20 y 0x7e inclusive.

De forma alternativa, cree una base de datos de claves utilizando la interfaz de usuario de **strmqikm** (iKeyman).

2. En sistemas UNIX and Linux, inicie una sesión con el usuario root. En los sistemas Windows, inicie la sesión como Administrador o como miembro del grupo MQM.
3. Inicie la interfaz de usuario ejecutando el mandato **strmqikm** .
4. En el menú **Archivo de base de datos de claves**, pulse **Nuevo**.

   Se abre la ventana Nuevo.
5. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
6. En el campo **Nombre de archivo**, escriba un nombre de archivo.

   Este campo ya contiene el texto `key.kdb`. Si el nombre de la raíz es `key`, no modifique el campo. Si ha especificado un nombre de raíz diferente, sustituya `key` por el nombre de raíz. Sin embargo, no debe cambiar la extensión `.kdb`.
7. En el campo **Ubicación**, escriba la vía de acceso.

   Por ejemplo:

   - Para un gestor de colas: `/var/mqm/qmgrs/QM1/ssl` (en sistemas UNIX and Linux) o `C:\ProgramData\IBM\MQ\qmgrs\QM1\ssl` (en sistemas Windows).

     La vía de acceso debe coincidir con el valor del atributo **SSLKeyRepository** del gestor de colas.
   - Para un cliente IBM MQ: `/var/mqm/ssl` (en sistemas UNIX and Linux) o `C:\mqm\ssl` (en sistemas Windows).
8. Pulse **Aceptar**.

   Se abre la ventana Solicitud de contraseña.
9. Escriba una contraseña en el campo **Contraseña** y vuelva a escribirla en el campo **Confirmar contraseña**.
10. Seleccione **Ocultar la contraseña en un archivo**.

    Nota: Si no oculta la contraseña, los intentos de iniciar los canales TLS fracasarán porque éstos no podrán obtener la contraseña necesaria para acceder al archivo de base de datos de claves.
11. Pulse **Aceptar**.

    Se abre la ventana Certificados personales.
12. Establezca los permisos de acceso tal como se describe en "Acceso y protección de los archivos de base de datos de claves en Windows" en la página 293 o "Acceso y protección de los archivos de base de datos de claves en sistemas UNIX and Linux" en la página 294.

Windows *Acceso y protección de los archivos de base de datos de claves en Windows*

Es posible que los archivos de base de datos de claves no tengan permisos de acceso adecuados. Debe establecer un acceso adecuado a estos archivos.

Establezca el control de acceso a los archivos *key*`.kdb`, *key*`.sth`, *key*`.crl` y *key*`.rdb`, donde *key* es el nombre de raíz de su base de datos de claves, para otorgar autorización a un conjunto de usuarios restringido.

Considere otorgar acceso del modo siguiente:

autorización total
BUILTIN\Administrators, NT AUTHORITY\SYSTEM y el usuario que creó los archivos de base de datos.

autorización de lectura
Para un gestor de colas, sólo el grupo mqm local. Con esto se presupone que el MCA se está ejecutando con un ID de usuario en el grupo mqm.

Para un cliente, el ID de usuario con el que se está ejecutando el proceso de cliente.

Linux UNIX *Acceso y protección de los archivos de base de datos de claves en sistemas UNIX and Linux*
Es posible que los archivos de base de datos de claves no tengan permisos de acceso adecuados. Debe establecer un acceso adecuado a estos archivos.

Para un gestor de colas, establezca los permisos en los archivos de bases de datos de claves de manera que el gestor de colas y los procesos de canales puedan leerlos cuando sea necesario pero que otros usuarios no puedan leerlos o modificarlos. Normalmente el usuario mqm necesita permisos de lectura. Si ha creado el archivo de bases de datos de claves iniciando sesión como usuario mqm, es posible que los permisos sean suficientes; si usted no era el usuario mqm sino otro usuario del grupo mqm, tal vez necesite otorgar permisos de lectura a otros usuarios del grupo mqm.

Igual que para un cliente, establezca los permisos en los archivos de las bases de datos de claves de manera que los procesos de la aplicación cliente puedan leerlos cuando sea necesario pero que otros usuarios no puedan leerlos o modificarlos. Normalmente el usuario con el que se ejecuta el proceso de cliente necesita permisos de lectura. Si ha creado el archivo de bases de datos de claves iniciando sesión como dicho usuario, es posible que los permisos sean suficientes; si usted no era el usuario cliente sino otro usuario de dicho grupo, tal vez necesite otorgar permisos de lectura a otros usuarios del grupo.

Establezca los permisos en los archivos *key*.kdb, *key*.sth, *key*.crl y *key*.rdb, donde *key* es el nombre de raíz de su base de datos de claves, en read y write para el propietario del archivo, y en read para el mqm o el grupo de usuarios del cliente (-rw-r-----).

ULW *Adición de certificados de CA predeterminados a un repositorio de claves vacío en UNIX, Linux, and Windows con GSKit 8.0*
Siga este procedimiento para añadir uno o varios de los certificados de autoridad emisora de certificados (CA) a un repositorio de claves vacío con GSKit versión 8.

En GSKit 7.0, el comportamiento al crear un nuevo repositorio de claves era añadir automáticamente un conjunto de certificados de CA predeterminados para las entidades emisoras de certificados utilizadas habitualmente. En GSKit versión 8, este comportamiento ha cambiado de modo que los certificados de CA ya no se añadan automáticamente al repositorio. El usuario ahora debe añadir manualmente certificados de CA en el repositorio de claves.

Utilización de `strmqikm`

Realice los pasos siguientes en la máquina a la que desea añadir el certificado de CA:

1. Inicie la GUI utilizando el mandato **strmqikm** (en UNIX, Linux, and Windows).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves al que desea añadir el certificado, por ejemplo, key.kdb.
6. Pulse **Abrir**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados de firmante**.
9. Pulse **Llenar**. Se abre la ventana de certificado de CA.

10. Los certificados de CA que están disponibles para ser agregados al repositorio se visualizan en una estructura de árbol jerárquica. Seleccione la entrada de nivel superior de la organización en cuyos certificados de CA desea confiar para ver la lista completa de certificados CA válidos.
11. Seleccione en la lista los certificados de CA en los que desee confiar y pulse **Aceptar**. Los certificados se añaden al repositorio de claves.

Utilización de la línea de mandatos

Utilice los mandatos siguientes para listar y, a continuación, añadir certificados de CA utilizando **runmqckm**:

- Emita el siguiente mandato para listar los certificados CA predeterminados junto con las organizaciones emisoras:

```
runmqckm -cert -listsigners
```

- Emita el siguiente mandato para añadir todos los certificados de CA para las organizaciones especificada en el campo *etiqueta* :

```
runmqckm -cert -populate -db filename -pw password -label label
```

donde:

-db *filename*	es el nombre de vía de acceso completo de la base de datos de claves.
-pw *password*	es la contraseña para la base de datos de claves.
-label *label*	es la etiqueta adherida al certificado.

Nota: Añadir un certificado de CA a un repositorio de claves da lugar a que results in IBM MQ confíe en todos los certificados personales firmados por el certificado de CA. Considere cuidadosamente en qué certificados de CA desea confiar y añada únicamente el conjunto de certificados de CA necesario para autenticar los clientes y los gestores. No se recomienda agregar todos los certificados de CA predeterminados a menos que sea un requisito de la política de seguridad.

ULW *Localizar el repositorio de claves para un gestor de colas en UNIX, Linux, and Windows*

Utilice este procedimiento para obtener la ubicación del archivo de base de datos de claves del gestor de colas.

Procedimiento

1. Visualice los atributos del gestor de colas, utilizando cualquiera de los mandatos MQSC siguientes:

```
DISPLAY QMGR ALL
DISPLAY QMGR SSLKEYR
```

   También puede visualizar los atributos del gestor de colas mediante IBM MQ Explorer o los mandatos PCF.
2. Examine la salida del mandato para localizar la vía de acceso y nombre de raíz del archivo de base de datos de claves.
   Por ejemplo,
   a. En UNIX and Linux: `/var/mqm/qmgrs/QM1/ssl/key`, donde `/var/mqm/qmgrs/QM1/ssl` es la vía de acceso y `key` es el nombre de raíz
   b. En Windows: *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl\key`, donde *MQ_INSTALLATION_PATH*`\qmgrs\QM1\ssl` es la vía de acceso y `key` es el nombre de raíz. *MQ_INSTALLATION_PATH* representa el directorio de alto nivel en el que está instalado IBM MQ.

ULW *Cambiar la ubicación del repositorio de claves para un gestor de colas en UNIX, Linux, and Windows*

Puede cambiar la ubicación del archivo de base de datos de claves del gestor de colas de diversas maneras, incluyendo el mandato MQSC ALTER QMGR.

Puede cambiar la ubicación del archivo de base de datos de claves del gestor de colas mediante el mandato MQSC ALTER QMGR para establecer el atributo de repositorio de claves del gestor de colas. Por ejemplo, en UNIX and Linux:

```
ALTER QMGR SSLKEYR('/var/mqm/qmgrs/QM1/ssl/MyKey')
```

El archivo de base de datos de claves tiene el nombre de archivo completo: `/var/mqm/qmgrs/QM1/ssl/MyKey.kdb`

En Windows:

```
ALTER QMGR SSLKEYR('C:\Archivos de programa\IBM\MQ\Qmgrs\QM1\ssl\Mykey')
```

El archivo de base de datos de claves tiene el nombre de archivo completo: `C:\Archivos de programa\IBM\MQ\Qmgrs\QM1\ssl\Mykey.kdb`

Atención: Asegúrese de que no incluye la extensión `.kdb` en el nombre de archivo en la palabra clave SSLKEYR, ya que el gestor de colas añade esta extensión automáticamente.

También puede alterar los atributos del gestor de colas mediante IBM MQ Explorer o los mandatos PCF.

Cuando se cambia la ubicación del archivo de base de datos de claves de un gestor de colas, los certificados no se transfieren desde la ubicación antigua. Si el archivo de base de datos de claves al que está accediendo ahora es un nuevo archivo de base de datos de claves, debe rellenarlo con los certificados de CA y personales que necesita, tal como se describe en “Importación de un certificado personal en un repositorio de claves en UNIX, Linux, and Windows” en la página 311.

ULW *Localizar el repositorio de claves para un IBM MQ MQI client en UNIX, Linux, and Windows*

La ubicación del repositorio de claves la proporciona la variable MQSSLKEYR, o se especifica en la llamada MQCONNX.

Examine la variable de entorno MQSSLKEYR para obtener la ubicación del archivo de base de datos de claves del IBM MQ MQI client. Por ejemplo:

```
echo $MQSSLKEYR
```

Compruebe también la aplicación, porque el nombre del archivo de base de datos de claves también se puede establecer en una llamada MQCONNX, según se describe en “Especificación de la ubicación del repositorio de claves para un IBM MQ MQI client en UNIX, Linux, and Windows” en la página 296. El valor establecido en una llamada MQCONNX altera temporalmente el valor de MQSSLKEYR.

ULW *Especificación de la ubicación del repositorio de claves para un IBM MQ MQI client en UNIX, Linux, and Windows*

No hay ningún repositorio de claves predeterminado para un IBM MQ MQI client. Puede especificar la ubicación del mismo de dos maneras. Asegúrese de que solamente puedan acceder al archivo de base de datos de claves los usuarios o administradores designados para impedir que se realice una copia no autorizada en otros sistemas.

Puede especificar la ubicación del archivo de base de datos de claves para el IBM MQ MQI client de dos formas:

- Estableciendo la variable de entorno MQSSLKEYR. Por ejemplo, en UNIX and Linux:

```
export MQSSLKEYR=/var/mqm/ssl/key
```

  El archivo de base de datos de claves tiene el nombre de archivo completo:

```
/var/mqm/ssl/key.kdb
```

  En Windows:

```
set MQSSLKEYR=C:\Archivos de programa\IBM\MQ\ssl\key
```

  El archivo de base de datos de claves tiene el nombre de archivo completo:

```
C:\Archivos de programa\IBM\MQ\ssl\key.kdb
```

  Nota: La extensión `.kdb` es una parte obligatoria del nombre de archivo pero no se incluye como parte del valor de la variable de entorno.
- Proporcionando la vía de acceso y el nombre de raíz del archivo de base de datos de claves en el campo *KeyRepository* de la estructura MQSCO cuando una aplicación realiza una llamada MQCONNX. Para obtener más información sobre la utilización de la estructura MQSCO en MQCONNX, consulte Visión general de MQSCO.

ULW *Cuándo entran en vigor los cambios en los certificados o en el almacén de certificados en UNIX, Linux, and Windows*

Cuando cambia los certificados de un almacén de certificados, o la ubicación del almacén de certificados, los cambios entran en vigor dependiendo del tipo de canal y de cómo se ejecuta el canal.

Los cambios efectuados en los certificados del archivo de base de datos de claves y en el atributo de repositorio de claves entran en vigor en las siguientes situaciones:

- Cuando un nuevo proceso de canal de salida individual ejecuta por primera vez un canal TLS.
- Cuando un nuevo proceso de canal de entrada individual TCP/IP recibe por primera vez una solicitud para iniciar un canal TLS.
- Cuando se emite el mandato MQSC REFRESH SECURITY TYPE(SSL) para renovar el entorno TLS.
- Para los procesos de la aplicación cliente, cuando se cierra la última conexión TLS del proceso. La siguiente conexión TLS recuperará los cambios del certificado.
- Para canales que se ejecutan como hebras de un proceso de agrupación de procesos (amqrmppa), cuando se inicia o se reinicia el proceso de agrupación de procesos y ejecuta por primera vez un canal TLS. Si el proceso de agrupación de procesos ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).
- Para canales que se ejecutan como hebras del iniciador de canal, cuando se inicia o se reinicia el iniciador de canal y ejecuta por primera vez un canal TLS. Si el proceso iniciador de canal ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).
- Para canales que se ejecutan como hebras de un escucha TCP/IP, cuando se inicia o se reinicia el escucha y recibe por primera vez una solicitud para iniciar un canal TLS. Si el escucha ya ha ejecutado un canal TLS y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).

También puede renovar el entorno TLS de IBM MQ utilizando IBM MQ Explorer o mandatos PCF.

ULW *Creación de un certificado personal autofirmado en UNIX, Linux, and Windows*

Puede crear un certificado autofirmado utilizando la interfaz gráfica de usuario de **strmqikm** (iKeyman), o desde la línea de mandatos utilizando **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

Nota: IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.

Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.

Para obtener más información sobre por qué utilizar certificados autofirmados, consulte Utilización de certificados autofirmados para la autenticación mutua de dos gestores de colas.

No todos los certificados digitales se pueden utilizar con todas las CipherSpecs. Asegúrese de crear un certificado que sea compatible con las CipherSpecs que necesita utilizar. IBM MQ da soporte a tres tipos distintos de CipherSpec. Si desea más detalles, consulte “Interoperatividad de Elliptic Curve y CipherSpecs RSA” en la página 46 en el tema “Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45.

Para utilizar las CipherSpecs de tipo 1 (aquellas con nombres que empiezan por ECDHE_ECDSA_) debe utilizar el mandato **runmqakm** para crear el certificado y debe especificar un parámetro de algoritmo de firma Elliptic Curve ECDSA; por ejemplo, **-sig_alg** EC_ecdsa_with_SHA384.

Consulte “opciones runmqckm y runmqakm en UNIX, Linux, and Windows” en la página 537 para obtener una lista de las opciones disponibles con el algoritmo hash **-sig_alg** .

Si está utilizando:

- GUI, consulte “Utilización de la interfaz de usuario de strmqikm” en la página 298
- Línea de mandatos, consulte “Utilización de la línea de mandatos” en la página 299

ULW *Utilización de la interfaz de usuario de **strmqikm***

Puede crear un certificado personal utilizando **strmqikm** (iKeyman) GUI.

Acerca de esta tarea

strmqikm no proporciona una opción compatible con FIPS. Si tiene que gestionar certificados TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Procedimiento

Realice los pasos siguientes para crear un certificado personal para el gestor de colas o IBM MQ MQI client utilizando la interfaz gráfica de usuario:

1. Inicie la GUI utilizando el mandato **strmqikm** .
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**.
   Se visualiza la ventana **Abrir** .
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves desde el que desea generar la solicitud; por ejemplo, key . kdb.
6. Pulse **Aceptar**.
   Se abre la ventana **Solicitud de contraseña**.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**.
   El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el menú **Crear**, pulse **Nuevo certificado autofirmado**. Se visualiza la ventana Crear nuevo certificado autofirmado.

9. En el campo **Etiqueta de clave**, escriba la etiqueta del certificado.

   La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.
10. Escriba o seleccione un valor para cualquier campo en el campo **Nombre distinguido** o bien cualquiera de los campos **Nombre alternativo de asunto**.
11. En los campos restantes, acepte los valores predeterminados o escriba o seleccione valores nuevos.

    Para obtener más información sobre los nombres distinguidos, consulte “Nombres distinguidos” en la página 11.
12. Pulse **Aceptar**.

    La lista **Certificados personales** muestra la etiqueta del certificado personal autofirmado que ha creado.

Qué hacer a continuación

Envíe una solicitud de certificado a una CA. En “Recepción de certificados personales en un repositorio de claves en UNIX, Linux, and Windows” en la página 305 encontrará más información.

ULW *Utilización de la línea de mandatos*

Puede crear un certificado personal desde la línea de mandatos utilizando los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Si tiene que gestionar certificados SSL o TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Procedimiento

Cree un certificado personal autofirmado utilizando el mandato **runmqckm** o **runmqakm** (GSKCapiCmd).

- Utilización de **runmqckm** en UNIX, Linux, and Windows:

```
runmqckm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days
 -sig_alg algorithm
```

  En lugar de `-dn distinguished_name`, puede utilizar `-san_dnsname DNS_names`, `-san_emailaddr email_addresses`o `-san_ipaddr IP_addresses`.

- Utilización de **runmqakm**:

```
runmqakm -cert -create -db filename -pw
password -label label
        -dn distinguished_name -size key_size
 -x509version version -expire days

        -fips -sig_alg algorithm
```

donde:

-db *nombrearchivo*
Especifica el nombre de archivo completo de una base de datos de claves CMS.

-pw *contraseña*
Especifica la contraseña para la base de datos de claves CMS.

-label *etiqueta*
Especifica la etiqueta de claves adjunta al certificado. La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Consulte

"Etiquetas de certificados digitales, descripción de los requisitos" en la página 26 para obtener más detalles.

-dn ***nombre_distinguido***
Especifica el nombre distinguido X.500 especificado entre comillas dobles. Se requiere al menos un atributo. Puede proporcionar varios atributos OU y DC.

Nota: Las herramientas **runmqckm** y **runmqakm** hacen referencia al atributo de código postal como POSTALCODE, no PC. Especifique siempre POSTALCODE en el parámetro **-dn** cuando utilice estos mandatos de gestión de certificados para solicitar certificados con un código postal.

-size ***tamaño_clave***
Especifica el tamaño de clave. Si utiliza **runmqckm**, el valor puede ser 512 o 1024. Si utiliza **runmqakm**, el valor puede ser 512, 1024 o bien 2048.

x509version ***versión***
Versión del certificado X.509 que se debe crear. El valor puede ser 1, 2 o 3. El valor predeterminado es 3.

-file ***nombrearchivo***
Especifica el nombre de archivo para la solicitud de certificado.

-expire ***Días***
Tiempo de caducidad del certificado en días. El valor predeterminado para un certificado es 365 días.

-fips
Especifica que el mandato se ejecuta en modalidad FIPS. Sólo se utiliza el componente ICC de FIPS y este componente debe inicializarse correctamente en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqaqm** falla.

-sig_alg
Para **runmqckm**, especifique el algoritmo de firma asimétrica utilizado para la creación del par de claves de la entrada. El valor puede ser, MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. El valor predeterminado es SHA1WithRSA.

-sig_alg
Para **runmqakm**, especifica el algoritmo de hash que se utiliza durante la creación de una solicitud de certificado. Este algoritmo de hash se utiliza para crear la firma asociada a la solicitud de certificado recién creada. El valor puede 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 o EC_ecdsa_with_SHA512. El valor predeterminado es SHA1WithRSA.

-san_dnsname ***nombres_DNS***
Especifica una lista delimitada por comas o delimitada por espacios de nombres DNS para la entrada que se está creando.

-san_emailaddr ***direcciones_correo_electrónico***
Especifica una lista delimitada por comas o una lista delimitada por espacios de direcciones de correo electrónicos para la entrada que se está creando.

-san_ipaddr ***direcciones_IP***
Especifica una lista delimitada por comas o por espacios de direcciones IP para la entrada que se está creando.

Qué hacer a continuación

Envíe una solicitud de certificado a una CA. En "Recepción de certificados personales en un repositorio de claves en UNIX, Linux, and Windows" en la página 305 encontrará más información.

ULW *Solicitud de un certificado personal en UNIX, Linux, and Windows*

Puede solicitar un certificado personal utilizando la interfaz gráfica de usuario (GUI) de **strmqikm** (iKeyman), o desde la línea de mandatos mediante los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Si tiene que gestionar certificados SSL o TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Acerca de esta tarea

Puede solicitar un certificado personal utilizando la GUI de **strmqikm** , o desde la línea de mandatos, sujeto a las consideraciones siguientes:

- IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.
- Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.
- No todos los certificados digitales se pueden utilizar con todas las CipherSpecs. Asegúrese de solicitar un certificado que sea compatible con las CipherSpecs que tiene que utilizar. IBM MQ da soporte a tres tipos distintos de CipherSpec. Si desea más detalles, consulte "Interoperatividad de Elliptic Curve y CipherSpecs RSA" en la página 46 en el tema "Certificados digitales y compatibilidad de CipherSpec en IBM MQ" en la página 45.
- Para utilizar las CipherSpecs de tipo 1 (con nombres que empiezan por ECDHE_ECDSA_), debe utilizar el mandato **runmqakm** para solicitar el certificado y debe especificar un parámetro de algoritmo de firma Elliptic Curve ECDSA; por ejemplo, **-sig_alg** EC_ecdsa_with_SHA384.

  Consulte "opciones runmqckm y runmqakm en UNIX, Linux, and Windows" en la página 537 para obtener una lista de las opciones disponibles con el algoritmo hash **-sig_alg** .
- Solo el mandato **runmqakm** proporciona una opción compatible con FIPS.
- Si utiliza hardware de cifrado, consulte "Solicitud de un certificado personal para el hardware PKCS #11" en la página 320.

Si está utilizando:

- GUI, consulte "Utilización de la interfaz de usuario de strmqikm" en la página 301
- Línea de mandatos, consulte "Utilización de la línea de mandatos" en la página 302

ULW *Utilización de la interfaz de usuario de **strmqikm***

Puede solicitar un certificado personal utilizando la interfaz gráfica de usuario (GUI) de **strmqikm** (iKeyman), o desde la línea de mandatos mediante los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Si tiene que gestionar certificados SSL o TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Acerca de esta tarea

strmqikm no proporciona una opción compatible con FIPS. Si tiene que gestionar certificados TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Procedimiento

Complete los pasos siguientes para solicitar un certificado personal utilizando la interfaz de usuario iKeyman:

1. Inicie la interfaz de usuario utilizando el mandato **strmqikm** .

2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**.

   Se abre la ventana **Abrir**.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves desde el que desea generar la solicitud; por ejemplo, `key.kdb`.
6. Pulse **Abrir**.

   Se abre la ventana **Solicitud de contraseña**.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**.

   El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el menú **Crear**, pulse **Nueva solicitud de certificado**. Se abre la ventana **Crear nueva clave y solicitud de certificado**.
9. En el campo **Etiqueta de clave**, escriba la etiqueta del certificado.

   La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.
10. Escriba o seleccione un valor para cualquier campo en el campo **Nombre distinguido** o bien cualquiera de los campos **Nombre alternativo de asunto**. En los campos restantes, acepte los valores predeterminados o escriba o seleccione valores nuevos.

    Para obtener más información sobre los nombres distinguidos, consulte “Nombres distinguidos” en la página 11.
11. En el campo **Especifique el nombre de un archivo en el que almacenar la solicitud de certificado**, acepte el valor predeterminado `certreq.arm` o escriba un valor nuevo con una vía de acceso completa.
12. Pulse **Aceptar**.

    Se visualiza una ventana de confirmación.
13. Pulse **Aceptar**.

    La lista **Solicitudes de certificados personales** muestra la etiqueta de la nueva solicitud de certificado personal que ha creado. La solicitud de certificado se almacenará en el archivo que ha seleccionado en el paso “11” en la página 302.
14. Solicite el nuevo certificado personal enviando el archivo a una entidad emisora de certificados (CA) o copiando el archivo en el formulario de solicitud en el sitio web de la CA.

ULW *Utilización de la línea de mandatos*

Puede solicitar un certificado personal desde la línea de mandatos utilizando los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd). Si tiene que gestionar certificados SSL o TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Procedimiento

Solicite un certificado personal utilizando el mandato **runmqckm** o **runmqakm** (GSKCapiCmd).

- Mediante **runmqckm**:

```
runmqckm -certreq -create -db filename -pw
password -label label
         -dn distinguished_name -size key_size
        -file filename -sig_alg algorithm
```

  En lugar de `-dn` *distinguished_name*, puede utilizar `-san_dsname` *DNS_names*, `-san_emailaddr` *email_addresses*o `-san_ipaddr` *IP_addresses*.

- Mediante **runmqakm**:

```
runmqakm -certreq -create -db filename -pw
password -label label
         -dn distinguished_name -size key_size
         -file filename -fips -sig_alg algorithm
```

donde:

-db ***nombrearchivo***
Especifica el nombre de archivo completo de una base de datos de claves CMS.

-pw ***contraseña***
Especifica la contraseña para la base de datos de claves CMS.

-label ***etiqueta***
Especifica la etiqueta de claves adjunta al certificado. La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado ibmwebspheremq con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Consulte "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26 para obtener más detalles.

-dn ***nombre_distinguido***
Especifica el nombre distinguido X.500 especificado entre comillas dobles. Se requiere al menos un atributo. Puede proporcionar varios atributos OU y DC.

Nota: Las herramientas **runmqckm** y **runmqakm** hacen referencia al atributo de código postal como POSTALCODE, no PC. Especifique siempre POSTALCODE en el parámetro **-dn** cuando utilice estos mandatos de gestión de certificados para solicitar certificados con un código postal.

-size ***tamaño_clave***
Especifica el tamaño de clave. Si utiliza **runmqckm**, el valor puede ser 512 o 1024. Si utiliza **runmqakm**, el valor puede ser 512, 1024 o bien 2048.

-file ***nombrearchivo***
Especifica el nombre de archivo para la solicitud de certificado.

-fips
Especifica que el mandato se ejecuta en modalidad FIPS. Cuando está en modalidad FIPS, el componente ICC utiliza los algoritmos que se han validado con FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

-sig_alg
Para **runmqckm**, especifique el algoritmo de firma asimétrica utilizado para la creación del par de claves de la entrada. El valor puede ser, MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. El valor predeterminado es SHA1WithRSA.

-sig_alg
Para **runmqakm**, especifica el algoritmo de hash que se utiliza durante la creación de una solicitud de certificado. Este algoritmo de hash se utiliza para crear la firma asociada a la solicitud de certificado recién creada. El valor puede 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 o EC_ecdsa_with_SHA512. El valor predeterminado es SHA1WithRSA.

-san_dnsname *nombres_DNS*
Especifica una lista delimitada por comas o delimitada por espacios de nombres DNS para la entrada que se está creando.

-san_emailaddr *direcciones_correo_electrónico*
Especifica una lista delimitada por comas o una lista delimitada por espacios de direcciones de correo electrónicos para la entrada que se está creando.

-san_ipaddr *direcciones_IP*
Especifica una lista delimitada por comas o por espacios de direcciones IP para la entrada que se está creando.

Qué hacer a continuación

Envíe una solicitud de certificado a una CA. En "Recepción de certificados personales en un repositorio de claves en UNIX, Linux, and Windows" en la página 305 encontrará más información.

ULW *Renovación de un certificado personal existente en UNIX, Linux, and Windows*

Puede renovar un certificado personal utilizando la interfaz gráfica de usuario (GUI) de **strmqikm** (iKeyman), o desde la línea de mandatos mediante los mandatos **runmqckm** (iKeycmd) o **runmqakm** (GSKCapiCmd).

Acerca de esta tarea

Si tiene un requisito de utilizar tamaños de clave mayores para sus certificados personales, no puede renovar un certificado existente. Debe sustituir la clave existente siguiendo los pasos descritos en "Solicitud de un certificado personal en UNIX, Linux, and Windows" en la página 301 para crear una nueva solicitud de certificado que utilice los tamaños de clave que necesite.

Un certificado personal tiene una fecha de caducidad, tras la cuál ya no se puede utilizar el certificado. Esta tarea describe la forma de renovar un certificado personal antes de que caduque.

Utilización de la interfaz de usuario de ***strmqikm***

Acerca de esta tarea

strmqikm no proporciona una opción compatible con FIPS. Si tiene que gestionar certificados TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Procedimiento

Realice los pasos siguientes para solicitar un certificado personal utilizando la interfaz de usuario de **strmqikm** :

1. Inicie la interfaz de usuario utilizando el mandato **strmqikm** en UNIX, Linux, and Windows.
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**.
   Se abre la ventana **Abrir**.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves desde el que desea generar la solicitud; por ejemplo, key.kdb.
6. Pulse **Abrir**.
   Se abre la ventana **Solicitud de contraseña**.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**.
   El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. Seleccione **Certificados personales** en el menú de selección desplegable y seleccione el certificado de la lista que desea renovar.

9. Pulse **Volver a crear solicitud ...** botón.

   Se abrirá una ventana para que especifique la información de nombre y ubicación de archivo.
10. En el campo **nombre de archivo**, acepte el valor predeterminado `certreq.arm` o escriba un valor nuevo, que incluya la vía de acceso de archivo completa.
11. Pulse **Aceptar**. La solicitud de certificado se almacenará en el archivo seleccionado en el paso "9" en la página 305.
12. Solicite el nuevo certificado personal enviando el archivo a una entidad emisora de certificados (CA) o copiando el archivo en el formulario de solicitud en el sitio web de la CA.

Utilización de la línea de mandatos

Procedimiento

Utilice los mandatos siguientes para solicitar un certificado personal utilizando el mandato **runmqckm** o bien **runmqakm**:

- Utilización de **runmqckm** en sistemas UNIX, Linux, and Windows :

```
runmqckm -certreq -recreate -db filename -pw
password -label label
-target filename
```

- Utilización de runmqakm:

```
runmqakm -certreq -recreate -db filename -pw
password -label label
-target filename
```

donde:

-db *nombrearchivo*
Especifica el nombre de archivo completo de una base de datos de claves CMS.

-pw *contraseña*
Especifica la contraseña para la base de datos de claves CMS.

-target *nombre_archivo*
Especifica el nombre de archivo para la solicitud de certificado.

Qué hacer a continuación

Una vez que haya recibido el certificado personal firmado de la entidad emisora de certificados, puede añadirlo a la base de datos de claves siguiendo los pasos descritos en "Recepción de certificados personales en un repositorio de claves en UNIX, Linux, and Windows" en la página 305.

ULW *Recepción de certificados personales en un repositorio de claves en UNIX, Linux, and Windows*

Utilice este procedimiento para recibir un certificado personal en el archivo de base de datos de claves. El repositorio de claves debe ser el mismo repositorio donde creó la solicitud de certificado.

Después de que la CA envíe un nuevo certificado personal, debe añadirlo al archivo de base de datos de claves desde donde ha generado la nueva solicitud de certificado. Si la CA envía el certificado como parte de un mensaje de correo electrónico, copie el certificado en un archivo aparte.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **runmqakm**. **strmqikm** no proporciona una opción compatible con FIPS.

Asegúrese de que el archivo de certificado que se va a importar dispone de permiso escrito para el usuario actual, y utilice el procedimiento siguiente para que un gestor de colas o un IBM MQ MQI client reciba un certificado personal en el archivo de base de datos de claves:

1. Inicie la GUI utilizando el mandato **strmqikm** (en Windows UNIX and Linux).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves al que desea añadir el certificado, por ejemplo, key.kdb.
6. Pulse **Abrir** y, a continuación, pulse **Aceptar**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**. Seleccione la vista de **Certificado personal**.
8. Pulse **Recibir**. Se abre la ventana Recibir certificado de un archivo.
9. Escriba el nombre de archivo de certificado y la ubicación del certificado personal nuevo, o pulse **Examinar** para seleccionar el nombre y la ubicación.
10. Pulse **Aceptar**. Si ya tiene un certificado personal en la base de datos de claves, se abre una ventana indicándole si desea establecer la clave que está añadiendo como clave predeterminada en la base de datos.
11. Pulse **Sí** o **No**. Se abre la ventana Entrar una etiqueta.
12. Pulse **Aceptar**. El campo **Certificados personales** muestra la etiqueta del nuevo certificado personal que ha añadido.

Utilización de la línea de mandatos

Para añadir un certificado personal a un archivo de base de datos de claves, utilice cualquiera de los mandatos siguientes:

- Mediante **runmqckm**:

```
runmqckm -cert -receive -file filename -db filename -pw password
         -format ascii
```

- Mediante **runmqakm**:

```
runmqakm -cert -receive -file filename -db filename -pw password -fips
```

donde:

-file *nombrearchivo*
Especifica el nombre de archivo completo del certificado personal.

-db *nombrearchivo*
Especifica el nombre de archivo completo de una base de datos de claves CMS.

-pw *contraseña*
Especifica la contraseña para la base de datos de claves CMS.

-format *ascii*
Especifica el formato del certificado. El valor puede ser `ascii` para datos ASCII codificados con Base64 o bien `binary` para datos DER binarios. El valor predeterminado es `ascii`.

-fips
Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

Si utiliza hardware de cifrado, consulte el apartado "Recepción de un certificado personal en el hardware PKCS #11" en la página 321.

ULW *Extracción de un certificado de CA de un repositorio de claves en UNIX, Linux, and Windows*

Siga este procedimiento para extraer un certificado de CA.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **runmqakm**. **strmqikm** (iKeyman) no proporciona una opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina desde la que desea extraer el certificado de CA:

1. Inicie la GUI utilizando el mandato **strmqikm** .
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves del que desea extraer, por ejemplo `key.kdb`.
6. Pulse **Abrir**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados de firmante** y seleccione el certificado que desea extraer.
9. Pulse **Extraer**. Se abre la ventana Extraer un certificado en un archivo.
10. Seleccione el **Tipo de datos** del certificado, por ejemplo, **Datos ASCII con codificación Base64** para un archivo con la extensión `.arm`.
11. Escriba el nombre del archivo de certificado y la ubicación del certificado donde desea guardar el certificado o pulse **Examinar** para seleccionar el nombre y la ubicación.
12. Pulse **Aceptar**. El certificado se escribirá en el archivo que ha especificado.

Utilización de la línea de mandatos

Utilice los mandatos siguientes para extraer un certificado de CA utilizando **runmqckm**:

- En UNIX, Linux, and Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

donde:

`-db` *filename*	es el nombre de vía de acceso completo de una base de datos de claves CMS.
`-pw` *password*	es la contraseña de la base de datos de claves CMS.
`-label` *label*	es la etiqueta adherida al certificado.
`-target` *filename*	es el nombre del archivo de destino.
`-format` *ascii*	es el formato del certificado. El valor puede ser `ascii` para datos ASCII codificados con Base64 o bien `binary` para datos DER binarios. El valor predeterminado es `ascii`.

`-fips`	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

ULW *Extracción de la parte pública de un certificado autofirmado de un repositorio de claves en UNIX, Linux, and Windows*

Siga este procedimiento para extraer la parte pública de un certificado autofirmado.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **runmqakm**. **strmqikm** (iKeyman) no proporciona una opción compatible con FIPS.

Realice los pasos siguientes en la máquina de la que desea extraer la parte pública de un certificado autofirmado:

1. Inicie la GUI utilizando el mandato **strmqikm** (en UNIX, Linux, and Windows).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves del que desea extraer el certificado, por ejemplo `key.kdb`.
6. Pulse **Aceptar**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados personales** y seleccione el certificado.
9. Pulse **Extraer certificado**. Se abre la ventana Extraer un certificado en un archivo.
10. Seleccione el **Tipo de datos** del certificado, por ejemplo, **Datos ASCII con codificación Base64** para un archivo con la extensión `.arm`.
11. Escriba el nombre del archivo de certificado y la ubicación del certificado donde desea guardar el certificado o pulse **Examinar** para seleccionar el nombre y la ubicación.
12. Pulse **Aceptar**. El certificado se escribirá en el archivo que ha especificado. Tenga en cuenta que cuando extraiga (en lugar de exportar) un certificado, sólo se incluirá la parte pública del certificado y, por lo tanto, la contraseña no es necesaria.

Utilización de la línea de mandatos

Utilice los mandatos siguientes para extraer la parte pública de un certificado autofirmado utilizando **runmqckm** o **runmqakm**:

- En UNIX, Linux, and Windows:

```
runmqckm -cert -extract -db filename -pw password -label label -target filename
         -format ascii
```

- Utilización de runmqakm:

```
runmqakm -cert -extract -db filename -pw password -label label
            -target filename -format ascii -fips
```

donde:

-db *filename*	es el nombre de vía de acceso completo de una base de datos de claves CMS.
-pw *password*	es la contraseña de la base de datos de claves CMS.
-label *label*	es la etiqueta adherida al certificado.
-target *filename*	es el nombre del archivo de destino.
-format *ascii*	es el formato del certificado. El valor puede ser `ascii` para datos ASCII codificados con Base64 o bien `binary` para datos DER binarios. El valor predeterminado es `ascii`.
-fips	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

ULW *Adición de un certificado de CA o la parte pública de un certificado autofirmado a un repositorio de claves en UNIX, Linux, and Windows*

Siga este procedimiento para añadir un certificado de CA o la parte pública de un certificado autofirmado al repositorio de claves.

Si el certificado que desea añadir se encuentra en una cadena de certificados, también debe añadir todos los certificados que están por encima suyo en la cadena. Debe añadir los certificados en orden estrictamente descendente, empezando por el raíz, seguido del certificado de CA inmediatamente debajo de éste en la cadena, y así sucesivamente.

Cuando las instrucciones siguientes hacen referencia a un certificado de CA, también se aplican a la parte pública de un certificado autofirmado.

Nota: Debe asegurarse de que el certificado esté en la codificación ASCII (UTF-8) o binaria (DER), ya que GSKit (IBM Global Secure Toolkit) no da soporte a certificados con otros tipos de codificación.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **runmqakm**. **strmqikm** no proporciona una opción compatible con FIPS.

Realice los pasos siguientes en la máquina a la que desea añadir el certificado de CA:

1. Inicie la GUI utilizando el mandato **strmqikm** (en sistemas UNIX, Linux y Windows).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves al que desea añadir el certificado, por ejemplo, `key.kdb`.
6. Pulse **Aceptar**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados de firmante**.
9. Pulse **Añadir**. Se abre la ventana Añadir certificado de CA desde un archivo.
10. Escriba el nombre del archivo de certificado y la ubicación donde se guardará el certificado personal o pulse **Examinar** para seleccionar el nombre y la ubicación.
11. Pulse **Aceptar**. Se abre la ventana Entrar una etiqueta.
12. En la ventana Entrar una etiqueta, escriba el nombre del certificado.
13. Pulse **Aceptar**. El certificado se añadirá a la base de datos de claves.

Utilización de la línea de mandatos

Para añadir un certificado de CA a una base de datos de claves, utilice cualquiera de los mandatos siguientes:

- Mediante **runmqckm**:

```
runmqckm -cert -add -db filename -pw password -label label
         -file filename -format ascii
```

- Mediante **runmqakm**:

```
runmqakm -cert -add -db filename -pw password -label label
         -file filename -format ascii -fips
```

donde:

-db *nombrearchivo*
Especifica el nombre de archivo completo de la base de datos de claves CMS.

-pw *contraseña*
Especifica la contraseña para la base de datos de claves CMS.

-label *etiqueta*
Especifica la etiqueta adherida al certificado.

-file *nombrearchivo*
Especifica el nombre del archivo que contiene el certificado.

-format *ascii*
Especifica el formato del certificado. El valor puede ser `ascii` para datos ASCII codificados con Base64 o bien `binary` para datos DER binarios. El valor predeterminado es `ascii`.

-fips
Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

ULW *Exportación de un certificado personal de un repositorio de claves en UNIX, Linux, and Windows*

Siga este procedimiento para exportar un certificado personal.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **runmqakm**. **strmqikm** (iKeyman) no proporciona una opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina desde la que desea exportar el certificado personal:

1. Inicie la GUI utilizando el mandato **strmqikm** (en Windows UNIX and Linux).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves desde el que desea exportar el certificado, por ejemplo `key.kdb`.
6. Pulse **Abrir**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados personales** y seleccione el certificado que desea exportar.
9. Pulse **Exportar/Importar**. Se abre la ventana Exportar/Importar.

10. Seleccione **Exportar clave**.
11. Seleccione el **Tipo de archivo de claves** del certificado que desea exportar, por ejemplo, **PKCS12**.
12. Escriba el nombre de archivo y la ubicación a la que desea exportar el certificado o pulse **Examinar** para seleccionar el nombre y la ubicación.
13. Pulse **Aceptar**. Se abre la ventana Solicitud de contraseña. Tenga en cuenta que cuando exporta (en lugar de extraer) un certificado, se incluyen las partes pública y privada del certificado. Por este motivo el archivo exportado se protege con contraseña. Cuando extraiga un certificado, sólo se incluirá la parte pública del certificado y, por lo tanto, la contraseña no es necesaria.
14. Escriba una contraseña en el campo **Contraseña** y vuelva a escribirla en el campo **Confirmar contraseña**.
15. Pulse **Aceptar**. El certificado se exporta al archivo que ha especificado.

Utilización de la línea de mandatos

Utilice los mandatos siguientes para exportar un certificado personal utilizando **runmqckm**:

- En UNIX, Linux, and Windows:

```
runmqckm -cert -export -db filename -pw password -label label -type cms
        -target filename -target_pw password -target_type pkcs12
```

donde:

-db *filename*	es el nombre de vía de acceso completo de la base de datos de claves de CMS.
-fips	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.
-pw *password*	es la contraseña de la base de datos de claves CMS.
-label *label*	es la etiqueta adherida al certificado.
-type *cms*	es el tipo de la base de datos.
-target *filename*	es el nombre de vía de acceso completo del archivo de destino.
-target_pw *password*	es la contraseña para cifrar el certificado.
-target_type *pkcs12*	es el tipo de certificado.

ULW *Importación de un certificado personal en un repositorio de claves en UNIX, Linux, and Windows*

Siga este procedimiento para importar un certificado personal.

Antes de importar un certificado personal en PKCS de formato #12 al archivo de base de datos de claves, primero debe añadir la cadena válida completa de emisión de certificados CA al archivo de base de datos de claves (consulte “Adición de un certificado de CA o la parte pública de un certificado autofirmado a un repositorio de claves en UNIX, Linux, and Windows” en la página 309).

Los archivos de PKCS #12 deben considerarse temporales y deben suprimirse después de su uso.

Utilización de strmqikm

Si tiene que gestionar certificados TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**. **strmqikm** no proporciona una opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina a la que desea importar el certificado personal:

1. Inicie la GUI utilizando el mandato **strmqikm** .
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves al que desea añadir el certificado, por ejemplo, key.kdb.
6. Pulse **Abrir**. Se visualiza la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En el campo **Contenido de la base de datos de claves**, seleccione **Certificados personales**.
9. Si hay certificados en la vista Certificados personales, siga estos pasos:
   a. Pulse **Exportar/Importar**. Se visualiza la ventana Exportar/Importar claves.
   b. Seleccione **Importar clave**.
10. Si no hay certificados en la vista Certificados personales, pulse **Importar**.
11. Seleccione el **Tipo de archivo de claves** del certificado que desea importar, por ejemplo PKCS12.
12. Escriba el nombre del archivo de certificado y la ubicación donde se guardará el certificado personal o pulse **Examinar** para seleccionar el nombre y la ubicación.
13. Pulse **Aceptar**. Se visualiza la ventana Solicitud de contraseña.
14. En el campo **Contraseña**, escriba la contraseña que se utilizó cuando se exportó el certificado.
15. Pulse **Aceptar**. Se visualizará la ventana Cambiar etiquetas. Puede cambiar las etiquetas de los certificados que se importan si, por ejemplo, ya existe un certificado con la misma etiqueta en la base de datos de claves de destino. El cambio de las etiquetas de certificados no tiene efecto en la validación de cadenas de certificados. Para asociar el certificado a un gestor de colas concreto o a un IBM MQ MQI client, IBM MQ utiliza el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de inicio de sesión de usuario de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.
16. Para cambiar una etiqueta, seleccione la etiqueta necesaria en la lista **Seleccionar una etiqueta para cambiar**. La etiqueta se copia en el campo de entrada **Entrar una nueva etiqueta**. Sustituya el texto de la etiqueta por el de la nueva etiqueta y pulse **Aplicar**.
17. El texto del campo de entrada **Entrar una nueva etiqueta** se copia en el campo **Seleccionar una etiqueta para cambiar**, sustituyendo a la etiqueta seleccionada originalmente y, por tanto, volviendo a etiquetar el certificado correspondiente.
18. Cuando haya cambiado todas las etiquetas que necesite cambiar, pulse **Aceptar**. La ventana Cambiar etiquetas se cerrará, y reaparecerá la ventana original de IBM Key Management con los campos **Certificados personales** y **Certificados de firmante** actualizados con los certificados etiquetados correctamente.
19. Se importa el certificado a la base de datos de claves.

Utilización de la línea de mandatos

Para importar un certificado personal utilizando **runmqckm**, utilice el mandato siguiente:

- En UNIX, Linux, and Windows:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label
```

donde:

`-file` *filename*	es el nombre de archivo completo del archivo que contiene el certificado PKCS #12.
`-pw` *password*	es la contraseña para el certificado PKCS #12.
`-type` *pkcs12*	es el tipo de archivo.
`-target` *filename*	es el nombre de la base de datos de claves del CMS de destino.
`-target_pw` *password*	es la contraseña de la base de datos de claves CMS.
`-target_type` *cms*	es el tipo de la base de datos especificada por `-target`
`-label` *label*	es la etiqueta del certificado a importar desde la base de datos de claves origen.
`-new_label` *label*	es la etiqueta que se asignará al certificado en la base de datos de destino. Si omite la opción `-new_label`, de forma predeterminada se utilizará el mismo valor que para la opción `-label`.
`-fips`	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

runmqckm no proporciona un mandato para cambiar las etiquetas de certificado directamente. Utilice los pasos siguientes para cambiar una etiqueta de certificado:

1. Exporte el certificado a un archivo PKCS #12 mediante el mandato **-cert -export**. Especifique la etiqueta de certificado existente para la opción `-label`.
2. Elimine la copia existente del certificado de la base de datos de claves original mediante el mandato **-cert -delete**.
3. Importe el certificado desde el archivo PKCS #12 utilizando el mandato **-cert -import** . Especifique la etiqueta antigua para la opción `-label` y la nueva etiqueta requerida para la opción `-new_label`. El certificado se volverá a importar a la base de datos de claves con la etiqueta requerida.

ULW *Importación de un certificado personal desde un archivo Microsoft.pfx*

Siga este procedimiento para importar desde un archivo Microsoft.pfx en UNIX, Linux, and Windows.

Un archivo .pfx puede contener dos certificados relativos a la misma clave. Uno es un certificado personal o certificado de sitio (que contiene una clave pública y una privada). El otro es un certificado de CA (de firmante) (que contiene sólo una clave pública). Estos certificados no pueden coexistir en el mismo archivo de base de datos de claves CMS, así que sólo se puede importar uno de ellos. Asimismo, el atributo "friendly name" o etiqueta se adjunta sólo al certificado de firmante.

El certificado personal se identifica mediante un identificador de usuario único (UUID) generado por el sistema. En este apartado se muestra la importación de un certificado personal de un archivo pfx al etiquetarlo con el friendly name asignado anteriormente al certificado de CA (firmante). Los certificados de CA (firmante) emisores ya deberían haberse añadido a la base de datos de claves de destino. Tenga en cuenta que los archivos de PKCS#12 deben considerarse temporales y deben suprimirse después de su uso.

Siga estos pasos para importar un certificado personal de una base de datos de claves pfx de origen:

1. Inicie la GUI utilizando el mandato **strmqikm** . Se visualiza la ventana IBM Key Management.
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se visualiza la ventana Abrir.
3. Seleccione un tipo de base de datos de claves de **PKCS12**.
4. **Se recomienda que haga una copia de seguridad de la base de datos pfx antes de realizar este paso.** Seleccione la base de datos de claves pfx que desea importar. Pulse **Abrir**. Se visualiza la ventana de solicitud de contraseña.

5. Entre la contraseña de la base de datos de claves y pulse **Aceptar**. Se visualiza la ventana IBM Key Management. La barra de título muestra el nombre del archivo de base de datos de claves pfx seleccionado, indicando que dicho archivo está abierto y está listo.
6. Seleccione **Certificados de firmante** de la lista. El atributo "friendly name" del certificado necesario se visualiza como etiqueta en el panel Certificados de firmante.
7. Seleccione la entrada de etiqueta y pulse **Suprimir** para eliminar el certificado de firmante. Se visualizará la ventana Confirmar.
8. Pulse **Sí**. La etiqueta seleccionada ya no se muestra en el panel Certificados de firmante.
9. Repita los pasos 6, 7 y 8 para todos los certificados de firmante.
10. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se visualiza la ventana Abrir.
11. Seleccione la base de datos de claves CMS de destino a la que se está importando el archivo pfx. Pulse **Abrir**. Se visualiza la ventana de solicitud de contraseña.
12. Entre la contraseña de la base de datos de claves y pulse **Aceptar**. Se visualiza la ventana IBM Key Management. La barra de título muestra el nombre del archivo de base de datos de claves seleccionado, indicando que dicho archivo está abierto y está listo.
13. Seleccione **Certificados personales** de la lista.
14. Si hay certificados en la vista Certificados personales, siga estos pasos:
    a. Pulse **Exportar/Importar claves**. Se visualiza la ventana Exportar/Importar claves.
    b. Seleccione **Importar** en Elegir tipo de acción.
15. Si no hay certificados en la vista Certificados personales, pulse **Importar**.
16. Seleccione el archivo PKCS12.
17. Especifique el nombre del archivo pfx tal como se utiliza en el paso 4. Pulse **Aceptar**. Se visualiza la ventana de solicitud de contraseña.
18. Especifique la misma contraseña que especificó cuando suprimió el certificado de firmante. Pulse **Aceptar**.
19. Se muestra la ventana Cambiar etiquetas (porque sólo debe haber un único certificado disponible para su importación). La etiqueta del certificado debe ser un UUID con el formato `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx`.
20. Para cambiar la etiqueta, seleccione el UUID en el panel **Seleccionar una etiqueta para cambiar:**. La etiqueta se replicará en el campo **Entrar una nueva etiqueta:**. Sustituya el texto de la etiqueta con el del friendly name que se suprimió en el Paso 7 y pulse **Aplicar**. El "friendly name" es el valor del atributo IBM MQ **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de inicio de sesión de usuario de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.
21. Pulse **Aceptar**. La ventana Cambiar etiquetas se eliminará, y reaparecerá la ventana original de IBM Key Management con los paneles de Certificados personales y Certificados de firmante actualizados con los certificados personales etiquetados correctamente.
22. El certificado personal pfx se ha importado a la base de datos (de destino).

No es posible cambiar una etiqueta de certificado utilizando **runmqckm** o **runmqakm**.

Utilización de la línea de mandatos

Para importar un certificado personal mediante **runmqckm** en UNIX, Linux, and Windows, utilice el mandato siguiente:

```
runmqckm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label -pfx
```

Para importar un certificado personal mediante **runmqakm**, utilice el siguiente mandato:

```
runmqakm -cert -import -file filename -pw password -type pkcs12 -target filename
-target_pw password -target_type cms -label label -fips -pfx
```

donde:

`-file` *filename*	es el nombre de archivo completo del archivo que contiene el certificado PKCS #12.
`-pw` *password*	es la contraseña para el certificado PKCS #12.
`-type` *pkcs12*	es el tipo de archivo.
`-target` *filename*	es el nombre de la base de datos de claves del CMS de destino.
`-target_pw` *password*	es la contraseña de la base de datos de claves CMS.
`-target_type` *cms*	es el tipo de la base de datos especificada por `-target`
`-label` *label*	es la etiqueta del certificado a importar desde la base de datos de claves origen.
`-new_label` *label*	es la etiqueta que se asignará al certificado en la base de datos de destino. Si omite la opción `-new_label`, de forma predeterminada se utilizará el mismo valor que para la opción `-label`.
`-fips`	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.
`-pfx`	indica el formato de archivo PFX.

runmqckm no proporciona un mandato para cambiar las etiquetas de certificado directamente. Utilice los pasos siguientes para cambiar una etiqueta de certificado:

1. Exporte el certificado a un archivo PKCS #12 mediante el mandato **-cert -export**. Especifique la etiqueta de certificado existente para la opción `-label`.
2. Elimine la copia existente del certificado de la base de datos de claves original mediante el mandato **-cert -delete**.
3. Importe el certificado desde el archivo PKCS #12 utilizando el mandato **-cert -import** . Especifique la etiqueta antigua para la opción `-label` y la nueva etiqueta requerida para la opción `-new_label`. El certificado se volverá a importar a la base de datos de claves con la etiqueta requerida.

ULW *Importación de un certificado personal desde un archivo PKCS #7*

Las herramientas **strmqikm** (iKeyman) y **runmqckm** (iKeycmd) no dan soporte a PKCS #7 (`.p7b`) . Utilice la herramienta **runmqckm** para importar certificados de un archivo PKCS #7 en UNIX, Linux, and Windows.

Utilice el siguiente mandato para añadir un certificado de CA de un archivo PKCS #7:

```
runmqckm -cert -add -db filename -pw password -type cms -file filename
-label label
```

`-db` *filename*	es el nombre de archivo completo de la base de datos de claves CMS.
`-pw` *password*	es la contraseña para la base de datos de claves.
`-type` *cms*	es el tipo de la base de datos de claves-
`-file` *filename*	es el nombre del archivo PKCS #7.

`-label` *label*	es la etiqueta que se asigna al certificado en la base de datos de destino. El primer certificado recibe la etiqueta especificada. Todos los demás certificados, si los hay, utilizan el nombre de asunto como etiqueta.

Utilice el siguiente mandato para importar un certificado personal desde un archivo 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*	es el nombre de archivo completo del archivo que contiene el certificado PKCS #7.
`-pw` *password*	es la contraseña para el certificado PKCS #7.
`-type` *pkcs7*	es el tipo de archivo.
`-target` *filename*	es el nombre de la base de datos de claves de destino.
`-target_pw` *password*	es la contraseña para la base de datos de claves de destino.
`-target_type` *cms*	es el tipo de la base de datos especificada por `-target`
`-label` *label*	es la etiqueta del certificado que se va a importar.
`-new_label` *label*	es la etiqueta que se asignará al certificado en la base de datos de destino. Si omite la opción `-new_label`, de forma predeterminada se utilizará el mismo valor que para la opción `-label`.

ULW *Supresión de un certificado de un repositorio de claves en UNIX, Linux, and Windows*

Utilice este procedimiento para suprimir certificados personales o de CA.

Utilización de `strmqikm`

Si tiene que gestionar certificados TLS de una forma que sea compatible con el estándar FIPS, utilice el mandato **`runmqakm`**. **`strmqikm`** (iKeyman) no proporciona una opción compatible con FIPS.

1. Inicie la GUI utilizando el mandato **`strmqikm`** (en UNIX, Linux, and Windows).
2. En el menú **Archivo de base de datos de claves**, pulse **Abrir**. Se abre la ventana Abrir.
3. Pulse **Tipo de base de datos de claves** y seleccione **CMS** (Certificate Management System).
4. Pulse **Examinar** para ir al directorio que contiene los archivos de base de datos de claves.
5. Seleccione el archivo de base de datos de claves del que desea suprimir el certificado, por ejemplo `key.kdb`.
6. Pulse **Abrir**. Se abre la ventana Solicitud de contraseña.
7. Escriba la contraseña que especificó al crear la base de datos de claves y pulse **Aceptar**. El nombre del archivo de base de datos de claves se visualiza en el campo **Nombre de archivo**.
8. En la lista desplegable, seleccione **Certificados personales** o **Certificados de firmante**
9. Seleccione el certificado que desea suprimir.
10. Si todavía no dispone de una copia del certificado y desea guardarla, pulse **Exportar/Importar** y expórtela (consulte el apartado “Exportación de un certificado personal de un repositorio de claves en UNIX, Linux, and Windows” en la página 310).
11. Con el certificado seleccionado, pulse **Suprimir**. Se abre la ventana Confirmar.
12. Pulse **Sí**. El campo **Certificados personales** ya no mostrará la etiqueta del certificado personal que ha suprimido.

Utilización de la línea de mandatos

Utilice los mandatos siguientes para suprimir un certificado utilizando **runmqckm**:

- En UNIX, Linux, and Windows:

```
runmqckm -cert -delete -db filename -pw password -label label
```

donde:

-db *filename*	es el nombre de archivo completo de una base de datos de claves CMS.
-pw *password*	es la contraseña de la base de datos de claves CMS.
-label *label*	es la etiqueta adherida al certificado personal.
-fips	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

ULW *Generación de contraseñas fuertes para la protección de repositorios de claves en UNIX, Linux, and Windows*

Puede generar contraseñas seguras para la protección de repositorios de claves mediante el mandato **runmqakm** (GSKCapiCmd).

Puede utilizar el mandato **runmqakm** con los siguientes parámetros para generar una contraseña segura:

```
runmqakm -random -create -length 14 -strong -fips
```

Al utilizar la contraseña generada en el parámetro **-pw** de mandatos de administración de certificados posteriores, incluya siempre la contraseña entre comillas dobles. En los sistemas UNIX and Linux, también debe utilizar un carácter de barra inclinada invertida para escapar los caracteres siguientes si aparecen en la serie de contraseña:

```
! \ " '
```

Al especificar la contraseña en respuesta a una solicitud de **runmqckm**, **runmqakm** o la GUI de **strmqikm** , no es necesario entrecomillar o escapar la contraseña. No es necesario porque el shell del sistema operativo no afecta a la entrada de entrada datos en estos casos.

ULW *Configuración del hardware de cifrado en UNIX, Linux, and Windows*

Puede configurar el hardware de cifrado para un gestor de colas o cliente de varias maneras.

Puede configurar hardware de cifrado para un gestor de colas en UNIX, Linux, and Windows mediante uno de los dos métodos siguientes:

- Utilice el mandato ALTER QMGR MQSC con el parámetro SSLCRYP, tal como se describe en ALTER QMGR.
- Utilice IBM MQ Explorer para configurar hardware de cifrado en el sistema UNIX, Linux o Windows. Para obtener más información, consulte la ayuda en línea.

Puede configurar hardware de cifrado para un cliente de IBM MQ en UNIX, Linux, and Windows mediante uno de los métodos siguientes:

- Establezca la variable de entorno MQSSLCRYP. Los valores permitidos para MQSSLCRYP son los mismos que para el parámetro SSLCRYP, tal como se describe en ALTER QMGR.

  Si utiliza la versión GSK_PKCS11 del parámetro SSLCRYP, la etiqueta de señal PKCS #11 debe coincidir con la etiqueta con la que ha configurado el hardware.

- Establezca el campo **CryptoHardware** de la estructura de opciones de configuración SSL, MQSCO, en una llamada MQCONNX. Si desea más información, consulte Visión general de MQSCO.

Si ha configurado hardware de cifrado que utiliza la interfaz PKCS #11 utilizando cualquiera de estos métodos, debe almacenar el certificado personal para utilizarlo en sus canales en el archivo de base de datos de claves del señal de cifrado que ha configurado. Este tema se describe en el apartado "Gestión de certificados en el hardware PKCS #11" en la página 318.

Gestión de certificados en el hardware PKCS #11

Puede gestionar certificados digitales en el hardware de cifrado que da soporte a la interfaz PKCS #11.

Acerca de esta tarea

Debe crear una base de datos de claves para preparar el entorno de IBM MQ, aunque no tenga la intención de almacenar en él certificados de CA (entidad emisora de certificados), pero almacenará los certificados en el hardware de cifrado. Es necesaria una base de datos de claves para que el gestor de claves haga referencia en el campo SSLKEYR o bien para que la aplicación cliente haga referencia a la variable de entorno MQSSLKEYR. Esta base de datos de claves también es necesaria si crea una solicitud de certificado.

Puede crear la base de datos de claves utilizando la línea de mandatos o utilizando la interfaz de usuario de **strmqikm** (iKeyman).

Procedimiento

Cree una base de datos de claves utilizando la línea de mandatos.

1. Ejecute uno de los mandatos siguientes:
   - Mediante **runmqckm**:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Mediante **runmqakm**:

     ```
     runmqakm -keydb -create -db filename -pw password -type cms
     -stash -fips -strong
     ```

   donde:

   -db *nombrearchivo*
   Especifica el nombre completo de una base de datos de claves CMS y debe tener una extensión de archivo de .kdb.

   -pw *contraseña*
   Especifica la contraseña para la base de datos de claves CMS.

   -type *cms*
   Especifica el tipo de base de datos. (Para IBM MQ, debe ser cms.)

   -stash
   Guarda la contraseña de la base de datos de claves en un archivo.

   -fips
   Especifica que el mandato se ejecuta en modalidad FIPS. Cuando está en modalidad FIPS, el componente ICC utiliza los algoritmos que se han validado con FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

   -strong
   Comprueba que la contraseña especificada cumple los requisitos mínimos de validez de contraseña. Los requisitos mínimos para una contraseña son los siguientes:

   - La contraseña debe tener una longitud mínima de 14 caracteres.

- La contraseña debe contener un mínimo de un carácter en minúsculas, un carácter en mayúsculas, y un dígito o un carácter especial. Los caracteres especiales incluyen el asterisco (*), el signo de dólar ($), el signo de número (#) y el signo de porcentaje (%). Un espacio se clasifica como un carácter especial.
- Cada carácter puede aparecer un máximo de tres veces en una contraseña.
- Dos es el número máximo de caracteres consecutivos que pueden ser idénticos.
- Todos los caracteres pertenecen al juego de caracteres ASCII imprimibles estándar dentro del rango entre 0x20 y 0x7e inclusive.

De forma alternativa, cree una base de datos de claves utilizando la interfaz de usuario de **strmqikm** (iKeyman).

2. En sistemas UNIX and Linux, inicie una sesión con el usuario root. En los sistemas Windows, inicie la sesión como Administrador o como miembro del grupo MQM.
3. Abra el archivo de propiedades de seguridad de Java, `java.security`.
   - En los sistemas UNIX and Linux, el archivo de propiedades de seguridad de Java se encuentra en el subdirectorio `java/jre64/jre/lib/security` del directorio de instalación de IBM MQ.
   - En los sistemas Windows, el archivo de propiedades de seguridad de Java se encuentra en el subdirectorio `java\jre\lib\security` del directorio de instalación de IBM MQ.

   Si todavía no está presente en el archivo, añada el proveedor de seguridad IBMPKCS11Impl. Por ejemplo, añadiendo la línea siguiente:

   ```
   security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl
   ```
4. Inicie la interfaz de usuario ejecutando el mandato **strmqikm** .
5. Pulse **Archivo de base de datos de claves** > **Abrir**.
6. Pulse **Tipo de base de datos de claves** y seleccione **PKCS11Direct**.
7. En el campo **Nombre de archivo**, escriba el nombre del módulo para gestionar el hardware de cifrado; por ejemplo, `PKCS11_API.so`.

   Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.
8. En el campo **Ubicación**, escriba la vía de acceso:
   - En los sistemas UNIX and Linux, podría ser `/usr/lib/pksc11`, por ejemplo.
   - En los sistemas Windows, puede escribir el nombre de la biblioteca; por ejemplo,`cryptoki`.

   Pulse **Aceptar**. Se abre la ventana Abrir señal de cifrado.
9. Seleccione la etiqueta de señal del dispositivo criptográfico que desea utilizar para almacenar los certificados.
10. En el campo **Contraseña de señal de cifrado**, escriba la contraseña que estableció al configurar el hardware de cifrado.
11. Si el hardware de cifrado tiene capacidad para contener los certificados de firmante necesarios para recibir o importar un certificado personal, borre ambos recuadros de selección de base de datos de claves secundarias y continúe desde el paso “15” en la página 320.

    Si necesita una base de datos de claves CMS para guardar los certificados del firmante, seleccione **Abrir archivo de base de datos existente** o bien **Crear nuevo archivo de base de datos de claves secundario**.
12. En el campo **Nombre de archivo**, escriba un nombre de archivo. Este campo ya contiene el texto `key.kdb`. Si el nombre de la raíz es `key`, no modifique el campo. Si ha especificado un nombre de raíz diferente, sustituya `key` por el nombre de raíz. Debe cambiar el sufijo `.kdb`.
13. En el campo **Ubicación**, escriba la vía de acceso, por ejemplo:

- Para un gestor de colas: `/var/mqm/qmgrs/QM1/ssl`
- Para un IBM MQ MQI client: `/var/mqm/ssl`

Pulse **Aceptar**. Se abre la ventana Solicitud de contraseña.

14. Escriba una contraseña.

    Si ha seleccionado **Abrir archivo de base de datos de claves secundarias existente** en el paso "11" en la página 319, escriba una contraseña en el campo **Contraseña**.

    Si ha seleccionado **Crear nuevo archivo de base de datos de claves secundarias** en el paso "11" en la página 319, realice los subpasos siguientes:

    a) Escriba una contraseña en el campo **Contraseña** y vuelva a escribirla en el campo **Confirmar contraseña**.

    b) Seleccione **Ocultar la contraseña en un archivo**. Tenga en cuenta que si no oculta la contraseña, los intentos de inicio de los canales TLS no se ejecutarán correctamente porque no podrán obtener la contraseña necesaria para acceder al archivo de base de datos de claves.

    c) Pulse **Aceptar**. Se abre una ventana confirmando que la contraseña se encuentra en el archivo `key.sth` (a menos que haya especificado un nombre de raíz diferente).

15. Pulse **Aceptar**. Se visualiza la sección de contenido de la base de datos de claves.

ULW *Solicitud de un certificado personal para el hardware PKCS #11*

Siga este procedimiento para un gestor de colas o un IBM MQ MQI client para solicitar un certificado personal para el hardware de cifrado.

Acerca de esta tarea

Esta tarea describe cómo utilizar la interfaz de usuario de **strmqikm** para solicitar un certificado personal. Si está utilizando la interfaz de línea de mandatos, consulte "Utilización de la línea de mandatos" en la página 302.

Nota: IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.

Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.

Procedimiento

Para solicitar un certificado personal desde la interfaz de usuario de **strmqikm** (iKeyman), realice los pasos siguientes:

1. Efectué estos pasos para trabajar con el hardware de cifrado. Consulte "Gestión de certificados en el hardware PKCS #11" en la página 318.
2. En el menú **Crear**, pulse **Nueva solicitud de certificado**.

   Se abre la ventana Crear nueva clave y solicitud de certificado.
3. En el campo **Etiqueta de clave**, escriba la etiqueta del certificado.

   La etiqueta es el valor del atributo **CERTLABL**, si éste está establecido o el valor predeterminado `ibmwebspheremq` con el nombre del gestor de colas o el ID de usuario de inicio de sesión de IBM MQ MQI client añadido, todo en minúsculas. Para obtener información detallada, consulte Etiquetas de certificados digitales.
4. Seleccione el **Tamaño de clave** y el **Algoritmo de firma** que necesite.
5. Escriba valores para **Nombre común** y **Organización** y seleccione un **País**. Para el resto de los campos opcionales puede aceptar los valores predeterminados o bien escribir o seleccionar valores nuevos.

   Tenga en cuenta que sólo puede suministrar un nombre en el campo **Unidad organizativa**. Para obtener más información sobre estos campos, consulte "Nombres distinguidos" en la página 11.

6. En el campo **Especifique el nombre de un archivo en el que almacenar la solicitud de certificado**, acepte el valor predeterminado `certreq.arm` o escriba un valor nuevo con una vía de acceso completa.
7. Pulse **Aceptar**.

   Se abrirá una ventana de confirmación.
8. Pulse **Aceptar**.

   La lista **Solicitudes de certificados personales** muestra la etiqueta de la nueva solicitud de certificado personal que ha creado. La solicitud de certificado se almacenará en el archivo que ha seleccionado en el paso "6" en la página 321.
9. Solicite el nuevo certificado personal enviando el archivo a una entidad emisora de certificados (CA) o copiando el archivo en el formulario de solicitud en el sitio web de la CA.

Recepción de un certificado personal en el hardware PKCS #11

Utilice este procedimiento para un gestor de colas o un IBM MQ MQI client para recibir un certificado personal en el hardware de cifrado.

Antes de empezar

Añada el certificado de CA de la CA que ha firmado el certificado personal. Añádalo al hardware de cifrado o a la base de datos de claves CMS secundaria. Haga esto antes de recibir el certificado firmado en el hardware de cifrado. Para añadir un certificado de CA a una base de datos de claves, siga el procedimiento de "Adición de un certificado de CA o la parte pública de un certificado autofirmado a un repositorio de claves en UNIX, Linux, and Windows" en la página 309.

Procedimiento

- Para recibir un certificado personal utilizando la interfaz de usuario de **strmqikm** (iKeyman), realice los pasos siguientes:
  a) Efectué estos pasos para trabajar con el hardware de cifrado. Consulte "Gestión de certificados en el hardware PKCS #11" en la página 318.
  b) Pulse **Recibir**. Se abre la ventana Recibir certificado de un archivo.
  c) Escriba el nombre de archivo de certificado y la ubicación del certificado personal nuevo, o pulse **Examinar** para seleccionar el nombre y la ubicación.
  d) Pulse **Aceptar**. Si ya tiene un certificado personal en su base de datos de claves, se abre una ventana en la que se le pregunta si desea establecer la clave que está añadiendo como la clave predeterminada de la base de datos.
  e) Pulse **Sí** o **No**. Se abre la ventana Entrar una etiqueta.
  f) Pulse **Aceptar**. La lista **Certificados personales** muestra la etiqueta del nuevo certificado personal que ha añadido. Esta etiqueta se forma añadiendo la etiqueta de la señal de cifrado delante de la etiqueta que ha proporcionado.
- Para recibir un certificado personal utilizando el mandato **runmqakm** (GSKCapiCmd), realice los pasos siguientes:
  a) Abra una ventana de mandatos configurada para su entorno.
  b) Reciba el certificado personal utilizando el mandato **runmqakm** (GSKCapiCmd):

```
runmqakm -cert -receive -file filename -crypto module_name
         -tokenlabel hardware_token -pw hardware_password
         -format cert_format -fips
         -secondaryDB filename -secondaryDBpw password
```

  donde:

  -file *nombrearchivo*
  Especifica el nombre de archivo completo del archivo que contiene el certificado personal.

-crypto *nombre_módulo*
Especifica el nombre completo de la biblioteca PKCS #11 proporcionada con el hardware de cifrado.

-tokenlabel *señal_hardware*
Especifica la etiqueta de señal del dispositivo criptográfico PKCS #11.

-pw *contraseña_hardware*
Especifica la contraseña para el acceso al hardware de cifrado.

-format *formato_cert*
Especifica el formato del certificado. El valor puede ser `ascii` para datos ASCII codificados con Base64 o bien `binary` para datos DER binarios. El valor predeterminado es ASCII.

-fips
Especifica que el mandato se ejecuta en modalidad FIPS. Cuando está en modalidad FIPS, el componente ICC utiliza los algoritmos que se han validado con FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.

-secondaryDB *nombrearchivo*
Especifica el nombre de archivo completo de la base de datos de claves CMS.

-secondaryDBpw *contraseña*
Especifica la contraseña para la base de datos de claves CMS.

MQ Appliance Trabajar con SSL/TLS en IBM MQ Appliance

IBM MQ Appliance tiene soporte de TLS (Transport Layer Security).

IBM MQ Appliance tiene diferentes mandatos para gestionar certificados. Para obtener información detallada sobre la gestión de certificados, consulte la documentación de IBM MQ Appliance, Gestión de certificados TLS

z/OS Trabajar con SSL/TLS en z/OS

Esta información describe cómo se configura y se trabaja con TLS (seguridad de la capa de transporte) en z/OS.

Cada tema incluye ejemplos de cómo realizar cada tarea utilizando RACF. Puede realizar tareas similares utilizando gestores de seguridad externos.

En z/OS, también debe establecer el número de subtareas de servidor que cada gestor de colas utiliza para procesar llamadas TLS, como se describe en “Definición del parámetro SSLTASKS en z/OS” en la página 323.

El soporte para TLS en z/OS forma parte integral del sistema operativo y se conoce como *SSL del sistema*. SSL del sistema forma parte del elemento de servicios de cifrado básicos de z/OS. Los miembros de Cryptographic Services Base se instalan en *pdsname*. Conjunto de datos particionados (PDS) SIEALNKE. Cuando instale SSL del sistema, asegúrese de que selecciona las opciones adecuadas para proporcionar las CipherSpecs que necesita.

z/OS *Requisitos de ID de usuario adicionales para TLS en z/OS*

Esta información describe los requisitos adicionales que su ID de usuario necesita para configurarse y funcionar con TLS en z/OS.

Asegúrese de que tiene todas las actualizaciones generales y de alto impacto (HIPER) en el sistema.

Asegúrese de que ha configurado los siguientes prerrequisitos:

- El ID de usuario *ssidCHIN* está correctamente definido en RACF y el ID de usuario *ssidCHIN* tiene acceso de lectura (READ) a los siguientes perfiles:
  - IRR.DIGTCERT.LIST
  - IRR.DIGTCERT.LISTRING

Estas variables están definidas en la clase FACILITY de RACF.

- El ID de usuario *ssidCHIN* es el propietario del conjunto de claves.
- El certificado personal del gestor de colas, si lo ha creado el mandato RACDCERT, se crea con un ID de usuario de tipo de certificado igual al del ID de usuario *ssidCHIN*.
- El iniciador de canal se recicla, o se emite el mandato **REFRESH SECURITY TYPE(SSL)** , para recoger los cambios que realice en el conjunto de claves.
- El procedimiento iniciador de canal de IBM MQ tiene acceso a la biblioteca de ejecución de SSL del sistema *pdsname*.SIEALNKE a través de la lista de enlaces, LPA o de una sentencia STEPLIB DD. Esta biblioteca debe estar autorizada para APF.
- El ID de usuario bajo cuya autoridad se ejecute el iniciador de canal se configura para utilizar UNIX System Services (USS), como se describe en la documentación de z/OS UNIX System Services Planning.

  Los usuarios que no desean que el iniciador de canal invoque UNIX System Services utilizando el UID invitado/predeterminado y el segmento OMVS, sólo necesitan modelar un segmento OMVS basado en el segmento predeterminado ya que el iniciador de canal no requiere permisos especiales y no se ejecuta dentro de UNIX como un superusuario.

z/OS *Definición del parámetro SSLTASKS en z/OS*

Utilice el mandato ALTER QMGR para establecer el número de subtareas de servidor para procesar llamadas TLS

Para utilizar canales TLS, asegúrese de que haya como mínimo dos subtareas de servidor, estableciendo el parámetro SSLTASKS del mandato ALTER QMGR. Por ejemplo:

```
ALTER QMGR SSLTASKS(5)
```

Para evitar problemas con la asignación de almacenamiento, no establezca atributo SSLTASKS en un valor superior a ocho en un entorno en el que no haya comprobación de CRL (Lista de revocación de certificados).

Si se usa la comprobación de CRL, un SSLTASK está retenido por el canal implicado para la duración de esa comprobación. Esto podría ser para un tiempo transcurrido significativo, mientras se contacta con el servidor LDAP relevante, porque cada SSLTASK es un bloque de control de tareas de z/OS.

Debe reiniciar el iniciador de canal si cambia el valor del atributo SSLTASKS.

z/OS *Configuración de un repositorio de claves en z/OS*

Configure un repositorio de claves en ambos extremos de la conexión. Asocie cada repositorio de claves a su gestor de colas.

Una conexión TLS requiere un *depósito de claves* en cada extremo de la conexión. Cada gestor de colas debe tener acceso a un repositorio de claves. Utilice el parámetro SSLKEYR del mandato ALTER QMGR para asociar un depósito de claves a un gestor de colas. Consulte “Repositorio de claves SSL/TLS” en la página 25 para obtener más información.

En z/OS, los certificados digitales se almacenan en un *conjunto de claves* que se gestiona mediante el gestor de seguridad externo (ESM). Estos certificados digitales tienen etiquetas, que asocian el certificado con un gestor de colas. TLS utiliza estos certificados con fines de autenticación. Todos los ejemplos siguientes utilizan mandatos RACF. Existen mandatos equivalentes para otros programas ESM.

En z/OS, IBM MQ utiliza el valor del atributo **CERTLABL**, si éste está establecido, o el valor predeterminado `ibmWebSphereMQ` con el nombre del gestor de colas añadido. Para obtener información detallada, consulte Etiquetas de certificados digitales.

El nombre del repositorio de claves de un gestor de colas es el nombre de un conjunto de claves de la base de datos RACF. Puede especificar el nombre de conjunto de claves antes o después de crearlo.

Utilice este procedimiento para crear un nuevo conjunto de claves para un gestor de colas:

1. Asegúrese de que tiene la autorización adecuada para emitir el mandato RACDCERT (consulte la publicación *SecureWay Security Server RACF Command Language Reference* para obtener más información).
2. Emita el mandato siguiente:

```
RACDCERT ID( userid1 ) ADDRING( ring-name )
```

   donde:

   - *idusuario1* es el ID de usuario del espacio de direcciones del iniciador de canal o el ID de usuario que será propietario del conjunto de claves (si el conjunto de claves es compartido).
   - *nombre_conjunto_claves* es el nombre que desea asignar al conjunto de claves. La longitud de este nombre puede tener un máximo de 237 caracteres. Este nombre es sensible a mayúsculas y minúsculas. Especifique *nombre_conjunto_claves* en mayúsculas para evitar problemas.

z/OS *Cómo poner certificados de CA a disposición de un gestor de colas en z/OS*

Una vez que haya creado el conjunto de claves, asocie al mismo cualquier certificado de CA relevante.

Si tiene el certificado de CA en un conjunto de datos, hay que añadir antes el certificado a la base de datos RACF con el mandato siguiente:

```
RACDCERT ID( userid1 ) ADD( input-data-set-name ) WITHLABEL( 'My CA' )
```

A continuación, para asociar un certificado de CA para My CA al conjunto de claves, utilice el mandato siguiente:

```
RACDCERT ID(userid1)
CONNECT(CERTAUTH LABEL('My CA') RING(ring-name) USAGE(CERTAUTH))
```

donde *idusuario1* es el ID de usuario del iniciador de canal o el propietario de un conjunto de claves compartido.

Para obtener más información sobre los certificados de CA, consulte “Certificados digitales” en la página 10.

z/OS *Localizar el repositorio de claves para un gestor de colas en z/OS*

Utilice este procedimiento para obtener la ubicación del conjunto de claves del gestor de colas.

1. Visualice los atributos del gestor de colas, utilizando cualquiera de los mandatos MQSC siguientes:

```
DISPLAY QMGR ALL
DISPLAY QMGR SSLKEYR
```

2. Examine la salida del mandato para obtener la ubicación del conjunto de claves.

z/OS *Especificación de la ubicación del repositorio de claves para un gestor de colas en z/OS*

Para especificar la ubicación del conjunto de claves del gestor de colas, utilice el mandato MQSC ALTER QMGR para establecer el atributo de repositorio de claves del gestor de colas.

Por ejemplo:

```
ALTER QMGR SSLKEYR(CSQ1RING)
```

si el conjunto de claves es propiedad del espacio de direcciones del iniciador de canal, o

```
ALTER QMGR SSLKEYR(userid1/CSQ1RING)
```

si es un conjunto de claves compartido, donde *idusuario1* es el ID del usuario propietario del conjunto de claves.

z/OS *Asignación de los derechos de acceso correctos al iniciador de canal en z/OS*

El iniciador de canal (CHINIT) necesita acceso al repositorio de claves y a ciertos perfiles de seguridad.

Otorgar al iniciador de canal (CHINIT) acceso para leer el depósito de claves

Si el ID de usuario CHINIT es el propietario del repositorio de claves, este ID de usuario necesita acceso de lectura al perfil IRR.DIGTCERT.LISTRING de la clase FACILITY, de lo contrario, necesita acceso de actualización. Otorgue acceso mediante el mandato PERMIT con ACCESS(UPDATE) o ACCESS(READ), según corresponda:

```
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID( userid ) ACCESS(UPDATE)
```

donde *idusuario* es el ID de usuario del espacio de direcciones del iniciador de canal.

Otorgar al iniciador de canal (CHINIT) acceso de lectura a los perfiles CSF* adecuados

Para que se pueda utilizar el soporte de hardware proporcionado a través de Integrated Cryptographic Service Facility (ICSF), asegúrese de que el ID de usuario de CHINIT tenga acceso de lectura a los perfiles CSF* adecuados en la clase CSFSERV, mediante el mandato siguiente:

```
PERMIT csf-resource CLASS(CSFSERV) ID( userid ) ACCESS(READ)
```

donde *recurso-csf* es el nombre del perfil CSF* e *idusuario* es el ID de usuario del espacio de direcciones del iniciador de canal.

Repita este mandato para cada uno de los siguientes perfiles CSF*:

- CSFDSG
- CSFDSV
- CSFPKD
- CSFPKE
- CSFPKI

Es posible que el ID de usuario CHINIT también necesite acceso de lectura a otros perfiles CSF*. Por ejemplo, si utiliza la especificación de cifrado ECDHE_RSA_AES_256_GCM_SHA384 , el ID de usuario CHINIT también necesita acceso de lectura a los siguientes perfiles CSF*:

- CSF1DVK
- CSF1GAV
- CSF1GKP
- CSF1SKE
- CSF1TRC
- CSF1TRD

Para obtener más información, consulte Requisitos de recursos de RACF CSFSERV.

Si las claves del certificado se almacenan en ICSF y su instalación ha establecido el control de acceso sobre las claves almacenadas en ICSF, asegúrese de que el ID de usuario CHINIT tenga acceso de lectura en el perfil de la clase CSFKEYS; para ello, emita el mandato siguiente:

```
PERMIT IRR.DIGTCERT. userid.* CLASS(CSFKEYS) ID( userid ) ACCESS(READ)
```

donde *idusuario* es el ID de usuario del espacio de direcciones del iniciador de canal.

Utilización de Integrated Cryptographic Service Facility (ICSF)

El iniciador de canal puede utilizar ICSF para generar un número aleatorio cuando se inicializa el algoritmo de protección por contraseña para enmascarar contraseñas que se envían a canales de cliente si no se está utilizando TLS.

Para obtener información adicional, consulte el apartado "Utilización de Integrated Cryptographic Service Facility (ICSF)" en la página 267

z/OS *Cuándo entran en vigor los cambios en los certificados o en el repositorio de claves en z/OS*

Los cambios entran en vigor cuando se inicia el iniciador de canal o se renueva el repositorio.

Concretamente, los cambios realizados en los certificados del conjunto de claves y en el atributo de repositorio de claves entran en vigor en las siguientes situaciones:

- Cuando el iniciador de canal se inicia o se reinicia.
- Cuando se emite el mandato REFRESH SECURITY TYPE(SSL) para renovar el contenido del repositorio de claves.

z/OS *Creación de un certificado personal autofirmado en z/OS*

Utilice este procedimiento para crear un certificado personal autofirmado.

1. Genere un certificado y un par de claves pública y privada mediante el mandato siguiente:

```
RACDCERT ID(userid2) GENCERT
SUBJECTSDN(CN('common-name')
           T('title')
           OU('organizational-unit')
           O('organization')
           L('locality')
           SP('state-or-province')
           C('country'))
WITHLABEL('label-name')
```

2. Asocie el certificado al conjunto de claves mediante el mandato siguiente:

```
RACDCERT ID(userid1)
CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
```

donde:

- *idusuario1* es el ID de usuario del espacio de direcciones del iniciador de canal o el propietario del conjunto de claves compartido.
- *idusuario2* es el ID de usuario asociado al certificado y debe ser el ID de usuario del espacio de direcciones del iniciador de canal.

  idusuario1 e *idusuario2* pueden ser el mismo ID.
- *nombre_conjunto_claves* es el nombre que ha asignado al conjunto de claves en "Configuración de un repositorio de claves en z/OS" en la página 323.

- *label-name* debe ser el valor del atributo IBM MQ **CERTLABL**, si está establecido, o el `ibmWebSphereMQ` predeterminado con el nombre del gestor de colas añadido. Para obtener información detallada, consulte Etiquetas de certificados digitales.

z/OS *Solicitud de un certificado personal en z/OS*

Solicite un certificado personal utilizando RACF.

Para solicitar un certificado personal, utilice RACF como se indica a continuación:

1. Cree un certificado personal autofirmado, como se muestra en el apartado "Creación de un certificado personal autofirmado en z/OS" en la página 326. Este certificado proporciona la solicitud con los valores de atributo del nombre distinguido.
2. Cree una solicitud de certificado PKCS #10 codificada en Base64 escrita en un conjunto de datos, con el mandato siguiente:

   ```
   RACDCERT ID(userid2) GENREQ(LABEL(' label_name ')) DSN(' output_data_set_name ')
   ```

   donde

   - *idusuario2* es el ID de usuario asociado al certificado y debe ser el ID de usuario del espacio de direcciones del iniciador de canal
   - *nombre_etiqueta* es la etiqueta que se utiliza cuando se crea el certificado autofirmado

   Consulte "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26 para obtener detalles.
3. Envíe el conjunto de datos a una Entidad emisora de certificados (CA) para solicitar un nuevo certificado personal.
4. Cuando la entidad emisora de certificados le devuelva el certificado firmado, añada de nuevo el certificado a la base de datos RACF, utilizando la etiqueta original, como se describe en "Adición de certificados personales a un repositorio de claves en z/OS" en la página 328.

z/OS *Creación de un certificado personal firmado por RACF*

RACF puede funcionar como una entidad emisora de certificados y emitir su propio certificado de CA.

Este apartado utiliza el término *certificado de firmante* para designar un certificado de CA emitido por RACF.

La clave privada del certificado de firmante debe estar en la base de datos de RACF antes de llevar a cabo el procedimiento siguiente:

1. Utilice el mandato siguiente para generar un certificado personal firmado por RACF, utilizando el certificado de firmante contenido en la base de datos RACF:

   ```
   RACDCERT ID(userid2) GENCERT
   SUBJECTSDN(CN('common-name')
              T('title')
              OU('organizational-unit')
              O('organization')
              L('locality')
              SP('state-or-province')
              C('country'))
   WITHLABEL('label-name')
   SIGNWITH(CERTAUTH LABEL('signer-label'))
   ```

2. Asocie el certificado al conjunto de claves mediante el mandato siguiente:

   ```
   RACDCERT ID(userid1)
   CONNECT(ID(userid2) LABEL('label-name') RING(ring-name) USAGE(PERSONAL))
   ```

donde:

- *idusuario1* es el ID de usuario del espacio de direcciones del iniciador de canal o el propietario del conjunto de claves compartido.
- *idusuario2* es el ID de usuario asociado al certificado y debe ser el ID de usuario del espacio de direcciones del iniciador de canal.

  idusuario1 e *idusuario2* pueden ser el mismo ID.
- *nombre_conjunto_claves* es el nombre que ha asignado al conjunto de claves en "Configuración de un repositorio de claves en z/OS" en la página 323.
- *nombre-etiqueta* debe ser el valor del atributo IBM MQ **CERTLABL**, si éste está establecido, o el valor predeterminado `ibmWebSphereMQ` con el nombre del gestor de colas o el grupo de compartición de colas añadido. Para obtener información detallada, consulte Etiquetas de certificados digitales.
- La *etiqueta_firmante* es la etiqueta de su propio certificado de firmante.

z/OS *Adición de certificados personales a un repositorio de claves en z/OS*

Utilice este procedimiento para añadir o importar un certificado personal a un conjunto de claves.

Después de que la entidad emisora de certificados le envíe un nuevo certificado personal, añádalo al conjunto de claves utilizando el siguiente procedimiento:

1. Añada el certificado a la base de datos RACF mediante el mandato siguiente:

```
RACDCERT ID( userid2 ) ADD( input-data-set-name ) WITHLABEL(' label-name ')
```

2. Asocie el certificado al conjunto de claves mediante el mandato siguiente:

```
RACDCERT ID( userid1 )
CONNECT(ID( userid2 ) LABEL(' label-name ') RING( ring-name ) USAGE(PERSONAL))
```

donde:

- *idusuario1* es el ID de usuario del espacio de direcciones del iniciador de canal o el propietario del conjunto de claves compartido.
- *idusuario2* es el ID de usuario asociado al certificado y debe ser el ID de usuario del espacio de direcciones del iniciador de canal.
- *nombre_conjunto_claves* es el nombre que ha asignado al conjunto de claves en "Configuración de un repositorio de claves en z/OS" en la página 323.
- *nombre-conjunto-datos-entrada* es el nombre del conjunto de datos que contiene el certificado de CA firmado. El conjunto de datos debe estar catalogado y no debe ser un PDS o un miembro de un PDS. El formato de registro (RECFM) que espera RACDCERT es VB. RACDCERT dinámicamente asigna y abre el conjunto de datos, y lee el certificado en forma de datos binarios.
- *nombre-etiqueta* es el nombre de etiqueta que se utilizó cuando se creó la solicitud original. Debe ser el valor del atributo IBM MQ **CERTLABL** , si está establecido, o el valor predeterminado `ibmWebSphereMQ` con el nombre del gestor de colas o grupo de compartición de colas añadido. Para obtener información detallada, consulte Etiquetas de certificados digitales.

z/OS *Exportación de un certificado personal de un repositorio de claves en z/OS*

Exporte el certificado mediante el mandato RACDCERT.

En el sistema desde el que desea exportar el certificado, utilice el mandato siguiente:

```
RACDCERT ID(userid2) EXPORT(LABEL('label-name'))
DSN(output-data-set-name) FORMAT(CERTB64)
```

donde:

- *idusuario2* es el ID de usuario con el que el certificado se ha añadido al conjunto de claves.
- El *nombre_etiqueta* es la etiqueta del certificado que desea extraer.

- El *nombre_conjunto_datos_salida* es el conjunto de datos en el que se coloca el certificado.
- CERTB64 es un certificado X.509 con codificación DER que tiene el formato Base64. Puede seleccionar un formato alternativo, por ejemplo:

  CERTDER
  Certificado X.509 con codificación DER en formato binario

  PKCS12B64
  El certificado PKCS #12 en formato Base64

  PKCS12DER
  El certificado PKCS #12 en formato binario

z/OS *Supresión de un certificado personal de un repositorio de claves en z/OS*

Suprima un certificado personal mediante el mandato RACDCERT.

Antes de suprimir un certificado personal, es posible que desee guardar una copia del mismo. Para copiar el certificado personal en un conjunto de datos antes de suprimirlo, siga el procedimiento que se indica en "Exportación de un certificado personal de un repositorio de claves en z/OS" en la página 328. A continuación, utilice el mandato siguiente para suprimir el certificado personal:

```
RACDCERT ID( userid2 ) DELETE(LABEL(' label-name '))
```

donde:

- *idusuario2* es el ID de usuario con el que el certificado se ha añadido al conjunto de claves.
- El *nombre_etiqueta* es el nombre del certificado que desea suprimir.

z/OS *Redenominación de un certificado personal en un repositorio de claves en z/OS*

Cambie el nombre de un certificado mediante el mandato RACDCERT.

Si no desea que se encuentre un certificado con una etiqueta específica, pero no desea suprimirlo, puede cambiarle el nombre temporalmente mediante el mandato siguiente:

```
RACDCERT ID( userid2 ) LABEL(' label-name ') NEWLABEL(' new-label-name ')
```

donde:

- *idusuario2* es el ID de usuario con el que el certificado se ha añadido al conjunto de claves.
- El *nombre-etiqueta* es el nombre del certificado cuyo nombre desea cambiar.
- *nombre-etiqueta-nuevo* es el nuevo nombre del certificado.

Esto puede resultar útil al probar la autenticación de cliente TLS.

z/OS *Asociación de un ID de usuario a un certificado digital en z/OS*

IBM MQ puede utilizar un ID de usuario asociado a un certificado RACF como un ID de usuario de canal. Asocie un ID de usuario a un certificado instalándolo bajo dicho ID de usuario, o utilizando un Filtro de nombre de certificado.

El método descrito en este tema es una alternativa al método independiente de la plataforma para asociar un ID de usuario a un certificado digital, el cual utiliza registros de autenticación de canal. Si desea más información sobre cómo los registros de autenticación de canal, consulte "Registros de autenticación de canal" en la página 49.

Cuando una entidad en un extremo de un canal TLS recibe un certificado de una conexión remota, la entidad solicita a RACF si hay un ID de usuario asociado a este certificado. La entidad utiliza el ID de usuario como el ID de usuario de canal. Si no hay ningún ID de usuario asociado al certificado, la entidad utiliza el ID de usuario con el que se ejecuta el iniciador de canal.

Asocie un ID de usuario a un certificado de una de las siguientes maneras:

- Instale el certificado en la base de datos RACF bajo el ID de usuario al que desea asociarlo, como se describe en “Adición de certificados personales a un repositorio de claves en z/OS” en la página 328.
- Utilice el CNF (filtro de nombres de certificados) para correlacionar el nombre distinguido del asunto o del emisor del certificado al ID de usuario, como se describe en “Configuración de un filtro de nombre de certificado en z/OS” en la página 330.

z/OS *Configuración de un filtro de nombre de certificado en z/OS*

Utilice el mandato RACDCERT para definir un filtro de nombre de certificado (CNF), que correlaciona un Nombre distinguido con un ID de usuario.

Realice los pasos siguientes para configurar un CNF.

1. Habilite las funciones de CNF utilizando el siguiente mandato. Para ello, necesita autorización de actualización en la clase DIGTNMAP.

   ```
   SETROPTS CLASSACT(DIGTNMAP) RACLIST(DIGTNMAP)
   ```

2. Defina el CNF. Por ejemplo:

   ```
   RACDCERT ID(USER1) MAP WITHLABEL('filter1') TRUST
   SDNFILTER('O=IBM.C=UK') IDNFILTER('O=ExampleCA.L=Internet')
   ```

   donde USER1 es el ID de usuario que se utilizará cuando:

   - El DN del asunto tenga como organización `IBM` y como país UK.
   - El DN del emisor tenga como organización `ExampleCA` y como localidad `Internet`.

3. Renueve las correlaciones de CNF:

   ```
   SETROPTS RACLIST(DIGTNMAP) REFRESH
   ```

Nota:

1. Si el certificado real está almacenado en la base de datos RACF, el ID de usuario con el que se instala tiene preferencia sobre el ID de usuario asociado a cualquier CNF. Si el certificado no está almacenado en la base de datos RACF, se utiliza el ID de usuario asociado al CNF coincidente más específico. Las coincidencias del DN del asunto consideran más específicas que las del DN del emisor.
2. Los cambios realizados en los CNF no se aplican hasta que se renuevan las correlaciones de CNF.
3. Un DN coincide con el filtro de DN de un CNF solamente si el filtro de DN es idéntico a la *parte menos significativa* del DN. La parte menos significativa del DN consta de los atributos que generalmente se listan en el extremo situado más a la derecha del DN, pero que aparece al principio del certificado.

   Por ejemplo, considere SDNFILTER `'O=IBM.C=UK'`. Un DN de sujeto `'CN=QM1.O=IBM.C=UK'` coincide con este filtro, pero un DN de sujeto `'CN=QM1.O=IBM.L=Hursley.C=UK'` no coincide con este filtro.

   La parte menos significativa de algunos certificados puede contener campos que no coincidan con el filtro de DN. Puede optar por excluir estos certificados especificando un patrón de DN en el patrón SSLPEER del mandato DEFINE CHANNEL.
4. Si el CNF coincidente más específico se define en RACF como NOTRUST, la entidad utiliza el usuario con el que se ejecuta el iniciador de canal.
5. RACF utiliza el carácter `'.'` como separador. IBM MQ utiliza una coma o un punto y coma.

Puede definir los CNF para asegurarse de que el servidor de entidad establece el ID de usuario de canal en el valor predeterminado, que es el ID de usuario con el que se ejecuta el iniciador de canal. Para cada certificado de CA del conjunto de claves asociado a la entidad, defina un CNF con un IDNFILTER que coincida exactamente con el DN del asunto de este certificado de CA. Esto garantiza que todos los certificados que pueda utilizar esta entidad coincidan como mínimo con uno de estos CNF. Eso es debido

a que todos los certificados de este tipo deben estar conectados al conjunto de claves asociado a la entidad o debe emitirlos una CA para la cual un certificado está conectado al conjunto de claves asociado a la entidad.

Consulte la publicación *SecureWay Security Server RACF Security Administrator's Guide* para obtener más información acerca de los mandatos que se utilizan para manipular los CNF.

z/OS *Definición de un canal emisor y una cola de transmisión en QMA en z/OS*

Utilice los mandatos **DEFINE CHANNEL** y **DEFINE QLOCAL** para configurar los objetos necesarios.

Procedimiento

En QMA, emita mandatos como los del siguiente ejemplo:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256) DESCR('Sender channel using TLS from QMA to QMB')

DEFINE QLOCAL(QMB) USAGE(XMITQ)
```

Resultados

Se crea un canal emisor, TO.QMB, y una cola de transmisión, QMB.

z/OS *Definición de un canal receptor en QMB en z/OS*

Utilice el mandato **DEFINE CHANNEL** para definir el objeto necesario.

Procedimiento

En QMB, emita un mandato como el del siguiente ejemplo:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS to QMB')
```

Resultados

Se crea un canal receptor, TO.QMB.

z/OS *Inicio del canal emisor en QMA en z/OS*

Si es necesario, inicie un programa de escucha y renueve la seguridad. A continuación, inicie el canal mediante el mandato **START CHANNEL**.

Procedimiento

1. Opcional: Si todavía no lo ha hecho, inicie un programa de escucha en QMB.

   El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal receptor cuando es necesario. Para obtener información sobre cómo iniciar un escucha, consulte Iniciar un escucha de canal.
2. Opcional: Si algún canal SSL/TLS se ha ejecutado anteriormente, emita el mandato REFRESH SECURITY TYPE(SSL).

   Esto garantiza que todos los cambios realizados en el repositorio de claves están disponibles.
3. Inicie el canal en QMA, utilizando el mandato START CHANNEL(TO.QMB).

Resultados

Se inicia el canal emisor.

z/OS *Intercambio de certificados autofirmados en z/OS*

Intercambie los certificados que ha extraído previamente. Si utiliza FTP, emplee el formato correcto.

Procedimiento

Transfiera la parte CA del certificado de QM1 al sistema de QM2 y viceversa, por ejemplo, por FTP.

Si transfiere los certificados mediante FTP, debe hacerlo en el formato correcto.

Transfiera los tipos de certificado siguientes en formato *binario*:

- X.509 binario codificado en DER
- PKCS #7 (certificados de CA)
- PKCS #12 (certificados personales)

Transfiera los siguientes tipos de certificado en formato ASCII:

- PEM (Privacy-Enhanced Mail)
- X.509 codificado en Base64

z/OS *Definición de un canal emisor y una cola de transmisión en QM1 en z/OS*

Utilice los mandatos **DEFINE CHANNEL** y **DEFINE QLOCAL** para configurar los objetos necesarios.

Procedimiento

En QM1, emita mandatos como los del ejemplo siguiente:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA) DESCR('Sender channel using TLS from QM1 to QM2')

DEFINE QLOCAL(QM2) USAGE(XMITQ)
```

Las CipherSpecs de cada extremo del canal deben ser la misma.

Sólo el parámetro SSLCIPH es obligatorio si desea que el canal utilice TLS. Consulte “CipherSpecs y CipherSuites en IBM MQ” en la página 40 para obtener información acerca de los valores permitidos para el parámetro SSLCIPH.

Resultados

Se crea un canal emisor, QM1.TO.QM2, y una cola de transmisión, QM2.

z/OS *Definición de un canal receptor en QM2 en z/OS*

Utilice el mandato **DEFINE CHANNEL** para definir el objeto necesario.

Procedimiento

En QM2, emita un mandato como el del ejemplo siguiente:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(TLS_RSA_WITH_AES_128_CBC_SHA256)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using TLS from QM1 to QM2')
```

El canal debe tener el mismo nombre que el canal emisor que ha definido en “Definición de un canal emisor y una cola de transmisión en QM1 en z/OS” en la página 332, y utilizar la misma especificación de cifrado (CipherSpec).

z/OS *Inicio del canal emisor en QM1 en z/OS*

Si es necesario, inicie un programa de escucha y renueve la seguridad. A continuación, inicie el canal mediante el mandato **START CHANNEL**.

Procedimiento

1. Opcional: Si todavía no lo ha hecho, inicie un programa de escucha en QM2.

   El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal receptor cuando es necesario. Para obtener información sobre cómo iniciar un escucha, consulte Iniciar un escucha de canal.
2. Opcional: Si se ha ejecutado anteriormente algún canal SSL/TLS, emita el mandato REFRESH SECURITY TYPE(SSL).

   Esto garantiza que todos los cambios realizados en el repositorio de claves están disponibles.
3. En QM1, inicie el canal, utilizando el mandato `START CHANNEL(QM1.TO.QM2)`.

Resultados

Se inicia el canal emisor.

z/OS *Renovación del entorno SSL o TLS en z/OS*

Renueve el entorno TLS en el gestor de colas QMA utilizando el mandato **REFRESH SECURITY** .

Procedimiento

En QMA, entre el siguiente mandato:

```
REFRESH SECURITY TYPE(SSL)
```

Esto garantiza que todos los cambios realizados en el repositorio de claves están disponibles.

z/OS *Permitir conexiones anónimas en un canal receptor en z/OS*

Utilice el mandato **ALTER CHANNEL** para hacer que la autenticación de cliente SSL o TLS sea opcional.

Procedimiento

En QMB, entre el siguiente mandato:

```
ALTER CHANNEL(TO.QMB) CHLTYPE(RCVR) SSLCAUTH(OPTIONAL)
```

z/OS *Inicio del canal emisor en QM1 en z/OS*

Si es necesario, inicie el iniciador de canal, inicie un programa de escucha y renueve la seguridad. A continuación, inicie el canal mediante el mandato **START CHANNEL**.

Procedimiento

1. Opcional: Si todavía no lo ha hecho, inicie el iniciador de canal.
2. Opcional: Si todavía no lo ha hecho, inicie un programa de escucha en QM2.

   El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal receptor cuando es necesario. Para obtener información sobre cómo iniciar un escucha, consulte Iniciar un escucha de canal.
3. Opcional: Si el iniciador de canal ya estaba en ejecución o si se han ejecutado anteriormente canales SSL/TLS, emita el mandato REFRESH SECURITY TYPE(SSL).

   Esto garantiza que todos los cambios realizados en el repositorio de claves están disponibles.
4. En QM1, inicie el canal, utilizando el mandato `START CHANNEL(QM1.TO.QM2)`.

Resultados

Se inicia el canal emisor.

z/OS *Inicio del canal emisor en QMA en z/OS*

Si es necesario, inicie el iniciador de canal, inicie un programa de escucha y renueve la seguridad. A continuación, inicie el canal mediante el mandato **START CHANNEL**.

Procedimiento

1. Opcional: Si todavía no lo ha hecho, inicie el iniciador de canal.
2. Opcional: Si todavía no lo ha hecho, inicie un programa de escucha en QMB.

   El programa de escucha está a la escucha de peticiones de red entrantes e inicia el canal receptor cuando es necesario. Para obtener información sobre cómo iniciar un escucha, consulte Iniciar un escucha de canal.
3. Opcional: Si el iniciador de canal ya se estaba ejecutando o si algún canal SSL/TLS se ha ejecutado anteriormente, emita el mandato `REFRESH SECURITY TYPE(SSL)`.

   Esto garantiza que todos los cambios realizados en el repositorio de claves están disponibles.
4. Inicie el canal en QMA, utilizando el mandato `START CHANNEL(TO.QMB)`.

Resultados

Se inicia el canal emisor.

z/OS *Modificación de la longitud de clave de curva elíptica en z/OS*

Cómo modificar la variable de entorno GSK_CLIENT_ECURVE_LIST, para establecer la lista de curvas elípticas o grupos soportados especificados por el cliente, como una serie que consta de uno o más valores de 4 caracteres en orden de preferencia para su uso.

Importante: Debe aplicar el arreglo en z/OS APAR OA61783 para permitir que determinadas curvas elípticas sean efectivas por el sistema operativo, cuando se utiliza TLS 1.0, TLS 1.1 y/o TLS 1.2 conexiones negociadas.

Puede establecer esta variable de entorno TLS en el JCL de inicio del iniciador de canal, utilizando la sentencia CEEOPTS DD:

```
CEEOPTS DD DSN=<dataset-name>,DISP=SHR
```

En el conjunto de datos al que se hace referencia anteriormente, especifique la lista que desea utilizar, por ejemplo:

```
ENVAR("GSK_CLIENT_ECURVE_LIST=002300240025")
```

Importante: No utilice esta sentencia CEEOPTS con datos en secuencia, ya que esto impide que se establezca la variable de entorno para todas las tareas TLS que utilizan dicha sentencia.

Asegúrese de que hace referencia a un conjunto de datos secuencial, o a un miembro de conjunto de datos particionado, para permitir que esto funcione cuando se utiliza un valor SSLTASKS mayor que uno.

También puede utilizar el equivalente analógico de servidor de GSK_CLIENT_ECURVE_LIST, que es GSK_SERVER_ALLOWED_KEX_ECURVES. Consulte Limitación de curvas elípticas de intercambio de claves para obtener más información.

Además, consulte la Tabla 5 en Definiciones de suite de cifrado para obtener una lista de curvas elípticas de 4 caracteres válidas y especificaciones de grupos soportadas.

La especificación predeterminada es `00210023002400250019`. Si TLS V1.3 está habilitado, `0029` `(x25519)` se añade al final de la lista predeterminada.

Identificación y autenticación de usuarios

Puede identificar y autenticar usuarios utilizando certificados X.509, la estructura MQCSP o en varios tipos de programa de salida de usuario.

Utilización de certificados X.509

Puede identificar y autenticar usuarios utilizando certificados x.509 con el mandato **CHLAUTH** y el parámetro **SSLPEER**. El parámetro **SSLPEER** especifica un filtro a utilizar para compararlo con el Nombre distinguido del sujeto del certificado del cliente o gestor de colas igual en el otro extremo del canal.

Para obtener más información sobre cómo utilizar el mandato **CHLAUTH** y el parámetro **SSLPEER** , consulte SET CHLAUTH.

Utilización de la estructura MQCSP

Debe especificar la estructura de parámetros de seguridad de conexión MQCSP en una llamada MQCONNX; esta estructura contiene un ID de usuario y una contraseña. Si es necesario, puede alterar el MQCSP en una salida de seguridad.

Nota: El gestor de autorizaciones sobre objetos (OAM) no utiliza la contraseña. No obstante, el OAM realiza un trabajo limitado con el ID de usuario, lo que puede considerarse una forma trivial de autenticación. Con estas comprobaciones, no es necesario adoptar otro ID de usuario, si utiliza esos parámetros en las aplicaciones.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de aplicación de cliente están protegidas adecuadamente, consulte “Protección por contraseña MQCSP” en la página 30.

Implementación de la identificación y autenticación en salidas de seguridad

El principal objetivo de una salida de seguridad es permitir que el MCA de cada extremo de un canal autentique su asociado. En cada extremo de un canal de mensajes, y en el extremo del servidor de un canal MQI, un MCA suele actuar en nombre del gestor de colas al que está conectado. En el extremo del cliente de un canal MQI, un MCA suele actuar en nombre del usuario de la aplicación cliente IBM MQ. En esta situación, la autenticación mutua realmente tiene lugar entre dos gestores de colas, o entre un gestor de colas y el usuario de una aplicación IBM MQ MQI client.

La salida de seguridad proporcionada (la salida de canal SSPI) ilustra cómo se puede implementar la autenticación mutua intercambiando señales de autenticación que genera, y posteriormente comprueba, un servidor de autenticación fiable como Kerberos. Para obtener más información, consulte “El programa de salida de canal SSPI en Windows” en la página 154.

La autenticación mutua también se puede implementar utilizando la tecnología de Infraestructura de claves públicas (PKI). Cada salida de seguridad genera algunos datos aleatorios, los firma utilizando la clave privada del gestor de colas o del usuario al que representa y envía los datos firmados a su asociado en un mensaje de seguridad. La salida de seguridad del asociado lleva a cabo la autenticación comprobando la firma digital mediante la clave pública del gestor de colas o usuario. Antes de intercambiar firmas digitales, es posible que las salidas de seguridad tengan que acordar el algoritmo para generar un resumen de mensaje, en el caso de que se pueda utilizar más de un algoritmo.

Cuando la salida de seguridad envía datos firmados a su asociado, también tiene que enviar algún medio de identificar el gestor de colas o usuario al que representa. Puede ser un Nombre distinguido o incluso un certificado digital. Si se envía un certificado digital, la salida de seguridad del asociado puede validar el certificado trabajando a través de una cadena de certificados hasta el certificado de CA raíz. Esto asegura la propiedad de la clave pública que se utiliza para comprobar la firma digital.

La salida de seguridad del asociado sólo puede validar un certificado digital si tiene acceso a un repositorio de claves que contiene los demás certificados de la cadena de certificados. Si no se envía un certificado digital correspondiente al gestor de colas o al usuario, debe haber uno disponible en el repositorio de claves al que la salida de seguridad del asociado tenga acceso. La salida de seguridad del asociado no puede comprobar la firma digital a no ser que encuentre la clave pública del firmante.

TLS (seguridad de la capa de transporte) utiliza técnicas PKI como las que se acaban de describir. Para obtener más información sobre cómo realiza TLS la autenticación, consulte “Conceptos de TLS (Transport Layer Security)” en la página 15.

Si no está disponible ningún servidor de autenticación fiable ni el soporte para PKI, se pueden utilizar otras técnicas. Una técnica común, que se puede implementar en salidas de seguridad, utiliza un algoritmo de clave simétrica.

Una de las salidas de seguridad, la salida A, genera un número aleatorio y lo envía en un mensaje de seguridad a su salida de seguridad asociada, la salida B. La salida B cifra el número utilizando su copia de una clave que sólo conocen las dos salidas de seguridad. La salida B envía el número cifrado a la salida A en un mensaje de seguridad con un segundo número aleatorio que ha generado la salida B. La salida A verifica que el primer número aleatorio se ha cifrado correctamente, cifra el segundo número aleatorio utilizando su copia de la clave y envía el número cifrado a la salida B en un mensaje de seguridad. Luego la salida B verifica que el segundo número aleatorio se ha cifrado correctamente. Durante este intercambio, si alguna de las salidas de seguridad no está satisfecha con la autenticidad de la otra, puede indicar al MCA que cierre el canal.

Una ventaja de esta técnica es que no se envía ninguna clave ni contraseña a través de la conexión de comunicaciones durante el intercambio. Una desventaja es que no proporciona una solución al problema de cómo distribuir la clave compartida de forma segura. Una solución a este problema se describe en "Implementación de confidencialidad en programas de salida de usuario" en la página 454. Una técnica parecida se utiliza en SNA para la autenticación mutua de dos LU cuando se vinculan para formar una sesión. La técnica se describe en "Autenticación a nivel de sesión" en la página 118.

Todas las técnicas anteriores para la autenticación mutua se pueden adaptar para proporcionar autenticación unidireccional.

Implementación de la identificación y autenticación en salidas de mensajes

Cuando una aplicación transfiere un mensaje a una cola, el campo *UserIdentifier* del descriptor de mensaje contiene un ID de usuario asociado a la aplicación. Sin embargo, no hay datos que se puedan utilizar para autenticar el ID de usuario. Estos datos se pueden añadir mediante una salida de mensajes en el extremo emisor de un canal y se pueden comprobar mediante una salida de mensajes en el extremo receptor del canal. Los datos de autenticación pueden ser una contraseña cifrada o una firma digital, por ejemplo.

Este servicio puede resultar más eficaz si se implementa a nivel de aplicación. El requisito básico es que el usuario de la aplicación que recibe el mensaje pueda identificar y autenticar al usuario de la aplicación que ha enviado el mensaje. Por lo tanto es natural considerar la implementación de este servicio a nivel de aplicación. Para obtener más información, consulte "Correlación de identidad en la salida de API y la salida cruzada de API" en la página 341.

Implementación de identificación y autenticación en la salida de API y la salida cruzada de API

En cuanto a un mensaje individual se refiere, la identificación y autenticación son un servicio en el que participan dos usuarios, el emisor y el receptor del mensaje. El requisito básico es que el usuario de la aplicación que recibe el mensaje pueda identificar y autenticar al usuario de la aplicación que ha enviado el mensaje. Tenga en cuenta que el requisito de autenticación es unidireccional y no bidireccional.

Dependiendo de cómo se implemente, es posible que los usuarios y sus aplicaciones necesiten una interfaz o deban interactuar con el servicio. Además, cuándo y cómo se utiliza el servicio dependerá de dónde están ubicados los usuarios y sus aplicaciones y de la naturaleza de las aplicaciones propiamente dichas. Por lo tanto, es natural considerar la implementación del servicio a nivel de aplicación en lugar de a nivel de enlace.

Si piensa implementar este servicio a nivel de enlace, es posible que deba tener en cuenta algunos aspectos como, por ejemplo, los siguientes:

- Cómo aplicará el servicio, en un canal de mensajes, solamente a los mensajes que lo requieren
- Cómo permitirá que los usuarios y las aplicaciones se comuniquen o interactúen con el servicio, si éste es un requisito

- Dónde invocará los componentes del servicio en una situación de varios saltos, en la que se envía un mensaje a través de más de un canal hasta llegar a su destino

A continuación se muestran algunos ejemplos de cómo se puede implementar el servicio de identificación y autorización a nivel de aplicación. El término *salida de API* significa una salida de API o una salida cruzada de API.

- Cuando una aplicación transfiere un mensaje a una cola, una salida de API puede obtener una señal de autenticación de un servidor de autenticación fiable, como Kerberos. La salida de API puede añadir esta señal a los datos de aplicación del mensaje. Cuando la aplicación receptora recupera el mensaje, una segunda salida de API puede solicitar al servidor de autenticación que autentique al emisor comprobando la señal.
- Cuando una aplicación transfiere un mensaje a una cola, se puede añadir una salida de API a los elementos siguientes de los datos de aplicación del mensaje:
  - El certificado digital del emisor
  - La firma digital del emisor

  Si se dispone de algoritmos diferentes para generar un resumen del mensaje, la salida de API puede incluir el nombre del algoritmo que ha utilizado.

  Cuando la aplicación receptora recupera el mensaje, una segunda salida de API puede realizar las comprobaciones siguientes:

  - La salida de API puede validar el certificado digital analizando la cadena de certificados hasta llegar al certificado de la CA raíz. Para hacerlo, la salida de API debe tener acceso al repositorio de claves que contiene los certificados restantes de la cadena de certificados. Esta comprobación asegura que el emisor, identificado mediante el Nombre distinguido, es el propietario genuino de la clave pública que contiene el certificado.
  - La salida de API puede comprobar la firma digital utilizando la clave pública que contiene el certificado. Esta comprobación autentica al emisor.

  El Nombre distinguido del emisor se puede enviar en lugar del certificado digital completo. En este caso, el repositorio de claves debe contener el certificado del emisor, de modo que la segunda salida de API pueda buscar la clave pública del emisor. Otra posibilidad es enviar todos los certificados de la cadena de certificados.
- Cuando una aplicación transfiere un mensaje a una cola, el campo *UserIdentifier* del descriptor de mensaje contiene un ID de usuario asociado a la aplicación. El ID de usuario se puede utilizar para identificar al emisor. Para habilitar la autenticación, una salida de API puede añadir algunos datos como, por ejemplo, una contraseña cifrada, a los datos de la aplicación que contiene el mensaje. Cuando la aplicación receptora recupera el mensaje, una segunda rutina de API puede autenticar el ID de usuario utilizando los datos que ha transportado el mensaje.

  Esta técnica puede considerarse suficiente en los mensajes que se originan en un entorno controlado y fiable, y en aquellas circunstancias en las que no se disponga de un servidor de autenticación fiable o de soporte para PKI.

Pluggable Authentication Method (PAM)

Linux UNIX

PAM es ahora común en las plataformas UNIX and Linux y proporciona un mecanismo general que oculta los detalles de autenticación de usuario de los servicios.

Se pueden utilizar reglas de autenticación distintas para servicios diferentes, configurando las reglas sin ningún cambio necesario para los propios servicios.

En “Utilización de PAM (Pluggable Authentication Method)” en la página 355 encontrará más información.

Usuarios privilegiados

Un usuario privilegiado es aquel que tiene autorización administrativa completa para IBM MQ.

Además de los usuarios listados en la tabla siguiente, hay ciertos objetos y autorizaciones a los que hay que prestar una atención especial a la hora de otorgar acceso, para garantizar la integridad y la seguridad del gestor de colas. Cuando se otorgue cualquiera de las autorizaciones siguientes, hay que prestar una atención especial:

- Cualquier autorización a un objeto SYSTEM
- Autorizaciones de administración para crear, alterar y suprimir objetos.

  z/OS En z/OS, esta es la autorización de seguridad de mandatos y la autorización de seguridad de recursos de mandato para emitir mandatos DEFINE, ALTER y DELETE.

  Multi En todas las demás plataformas, estas autorizaciones son autorizaciones de administración como, por ejemplo, `+crt`, `+chg` y `+dlt`.
- Autorización de administración para borrar colas.

  z/OS En z/OS, esta es la autorización de seguridad de mandatos y la autoridad de seguridad de recursos de mandato para emitir mandatos CLEAR.

  Multi En todas las demás plataformas, esta autorización es `+clr`.
- Las autorizaciones de administración para detener canales, restituir o confirmar mensajes.

  z/OS En z/OS, esta autorización es de seguridad de mandatos y la autorización de seguridad de recursos de mandatos para emitir mandatos como, por ejemplo, RESET CHANNEL, START CHANNEL y STOP CHANNEL.

  Multi En todas las demás plataformas, estas autorizaciones son `+ctrl` y `+ctrlx`.
- La autorización MQI de usuario alternativo que permite a las aplicaciones escalar privilegios para comprobaciones de autorización.

  z/OS En z/OS, esta autorización es cualquier autorización otorgada a los perfiles de seguridad de usuario alternativo.

  Multi En todas las demás plataformas, esta autorización es `+altusr`.
- Las autorizaciones de contexto que permiten a las aplicaciones cambiar el contexto de seguridad de los mensajes.

  z/OS En z/OS, esta autorización es cualquier autorización otorgada a los perfiles de seguridad de contexto.

  Multi En todas las demás plataformas, estas autorizaciones son `+setall` y `+setid`.

Como regla general, a las aplicaciones de mensajería solo se les debería otorgar las autorizaciones MQI en las colas o temas que sean necesarios. Los canales MCA que ejecutan con un MCAUSER sin privilegios y algunos otros tipos de aplicaciones especiales como, por ejemplo, manejadores de colas de mensajes no entregados, pueden requerir autorizaciones adicionales que no suelen otorgarse a las aplicaciones para que funcionen correctamente.

Tabla 67. Usuarios privilegiados por plataforma

Plataforma	**Usuarios privilegiados**
Sistemas Windows	• SYSTEM • Miembros del grupo mqm • Miembros del grupo Administradores
Sistemas UNIX and Linux	• Miembros del grupo mqm

Tabla 67. Usuarios privilegiados por plataforma (continuación)	
Plataforma	**Usuarios privilegiados**
IBM i Sistemas IBM i	• Los perfiles qmqm y qmqmadm • Todos los miembros del grupo qmqmadm • Cualquier usuario definido con el valor *ALLOBJ
z/OS	El ID de usuario bajo los que se ejecutan el iniciador de canal, el gestor de colas y los espacios de direcciones de seguridad de mensajería avanzada. Estos ID de usuario no tienen automáticamente las autorizaciones administrativas completas para IBM MQ, pero se consideran privilegiados debido al nivel de acceso que normalmente suelen recibir estos ID de usuario.

Identificación y autenticación de usuarios utilizando la estructura MQCSP

Puede especificar la estructura de parámetros de seguridad de conexión MQCSP en una llamada MQCONNX.

La estructura de parámetros de seguridad de conexión MQCSP contiene un ID de usuario y una contraseña, que el servicio de autorización puede utilizar para identificar y autenticar el usuario.

Puede alterar el MQCSP en una salida de seguridad.

Aviso: En algunos casos, la contraseña en la estructura MQCSP para una aplicación cliente se enviará por una red en texto sin formato. Para asegurarse de que las contraseñas de aplicación de cliente están protegidas adecuadamente, consulte “Protección por contraseña MQCSP” en la página 30.

Relación entre los valores de MQCSP y AdoptCTX

IBM MQ siempre autentica las credenciales pasadas a través de la estructura MQCSP a menos que la característica de autenticación de conexión no esté habilitada. Una vez que las credenciales se han autenticado correctamente, IBM MQ intenta adoptar el ID de usuario para futuras comprobaciones de autorización a menos que ADOPTCTX no esté habilitado.

IBM MQ tiene un límite en la longitud de los ID de usuario que puede utilizar para comprobaciones de autorización. Estos límites se detallan en “ID de usuario” en la página 83. Al adoptar un ID de usuario pasado a través de la estructura MQCSP, IBM MQ se comporta de forma diferente, en función de otras opciones de configuración:

- Cuando se utiliza la autenticación de conexión LDAP, IBM MQ recupera el valor del campo establecido en SHORTUSR del registro LDAP del usuario de dicho usuario y adopta dicho ID de usuario.

  Por ejemplo, si SHORTUSR se establece en `'CN'` y un registro LDAP lista un usuario como `'CN=Test,SN=MQ,O=IBM,C=UK'`, se utiliza el ID de usuario `Test`.
- Cuando se utiliza la autenticación de conexión de sistema operativo o la autenticación PAM, si ADOPTCTX es YES, el ID de usuario pasado a través de la estructura MQCSP se trunca para cumplir el límite de ID de usuario de 12 caracteres de IBM MQ cuando se adopta como contexto de conexión.

  Si **ChlAuthEarlyAdopt** está habilitado, el truncamiento se produce después de que se hayan autenticado las credenciales de usuario.

  Si **ChlAuthEarlyAdopt** no está habilitado, el truncamiento se produce antes de la adopción. En Windows, si el usuario se proporciona con el formato `user@domain`, esto significa que el truncamiento puede dar como resultado una especificación de dominio que no es válida cuando el usuario tiene menos de 12 caracteres.

Por ejemplo, si se proporciona un usuario `ibmmq@windowsdomain` a través de MQCSP, se trunca en `ibmmq@window` en este escenario. Esto da como resultado el siguiente error:

```
AMQ8074W: La autorización ha fallado porque el SID 'SID' no coincide con la entidad
'ibmmq@window'
```

Sobre esta base, si pasa un ID de usuario de más de 12 caracteres, como un ID de usuario de dominio de Windows con el formato `user@domain`, a través de MQCSP debe configurar **ChlAuthEarlyAdopt**=*Y* en el archivo qm.ini para evitar este error.

De forma alternativa, utilice ADOPTCTX (NO) en la configuración CONNAUTH AUTHINFO y utilice un enfoque alternativo como, por ejemplo, una regla CHLAUTH USERMAP, una salida de seguridad o el valor MCAUSER del objeto de canal para establecer el ID de usuario para el canal.

Implementación de la identificación y autenticación en salidas de seguridad

Puede utilizar una salida de seguridad para implementar autenticación unidireccional o mutua

El principal objetivo de una salida de seguridad es permitir que el MCA de cada extremo de un canal autentique su asociado. En cada extremo de un canal de mensajes, y en el extremo del servidor de un canal MQI, un MCA suele actuar en nombre del gestor de colas al que está conectado. En el extremo del cliente de un canal MQI, un MCA suele actuar en nombre del usuario de la aplicación IBM MQ MQI client. En esta situación, la autenticación mutua realmente tiene lugar entre dos gestores de colas, o entre un gestor de colas y el usuario de una aplicación IBM MQ MQI client.

La salida de seguridad proporcionada (la salida de canal SSPI) ilustra cómo se puede implementar la autenticación mutua intercambiando señales de autenticación que genera, y posteriormente comprueba, un servidor de autenticación fiable como Kerberos. Para obtener más información, consulte “El programa de salida de canal SSPI en Windows” en la página 154.

La autenticación mutua también se puede implementar utilizando la tecnología de Infraestructura de claves públicas (PKI). Cada salida de seguridad genera algunos datos aleatorios, los firma utilizando la clave privada del gestor de colas o del usuario al que representa y envía los datos firmados a su asociado en un mensaje de seguridad. La salida de seguridad del asociado lleva a cabo la autenticación comprobando la firma digital mediante la clave pública del gestor de colas o usuario. Antes de intercambiar firmas digitales, es posible que las salidas de seguridad tengan que acordar el algoritmo para generar un resumen de mensaje, en el caso de que se pueda utilizar más de un algoritmo.

Cuando la salida de seguridad envía datos firmados a su asociado, también tiene que enviar algún medio de identificar el gestor de colas o usuario al que representa. Puede ser un Nombre distinguido o incluso un certificado digital. Si se envía un certificado digital, la salida de seguridad del asociado puede validar el certificado trabajando a través de una cadena de certificados hasta el certificado de CA raíz. Esto asegura la propiedad de la clave pública que se utiliza para comprobar la firma digital.

La salida de seguridad del asociado sólo puede validar un certificado digital si tiene acceso a un repositorio de claves que contiene los demás certificados de la cadena de certificados. Si no se envía un certificado digital correspondiente al gestor de colas o al usuario, debe haber uno disponible en el repositorio de claves al que la salida de seguridad del asociado tenga acceso. La salida de seguridad del asociado no puede comprobar la firma digital a no ser que encuentre la clave pública del firmante.

TLS (seguridad de la capa de transporte) utiliza técnicas PKI como las que se acaban de describir. Para obtener más información sobre cómo SSL lleva a cabo la autenticación, consulte “Conceptos de TLS (Transport Layer Security)” en la página 15.

Si no está disponible ningún servidor de autenticación fiable ni el soporte para PKI, se pueden utilizar otras técnicas. Una técnica común, que se puede implementar en salidas de seguridad, utiliza un algoritmo de clave simétrica.

Una de las salidas de seguridad, la salida A, genera un número aleatorio y lo envía en un mensaje de seguridad a su salida de seguridad asociada, la salida B. La salida B cifra el número utilizando su copia de una clave que sólo conocen las dos salidas de seguridad. La salida B envía el número cifrado a la salida A en un mensaje de seguridad con un segundo número aleatorio que ha generado la salida B. La salida A verifica que el primer número aleatorio se ha cifrado correctamente, cifra el segundo número aleatorio utilizando su copia de la clave y envía el número cifrado a la salida B en un mensaje de seguridad.

Luego la salida B verifica que el segundo número aleatorio se ha cifrado correctamente. Durante este intercambio, si alguna de las salidas de seguridad no está satisfecha con la autenticidad de la otra, puede indicar al MCA que cierre el canal.

Una ventaja de esta técnica es que no se envía ninguna clave ni contraseña a través de la conexión de comunicaciones durante el intercambio. Una desventaja es que no proporciona una solución al problema de cómo distribuir la clave compartida de forma segura. Una solución a este problema se describe en "Implementación de confidencialidad en programas de salida de usuario" en la página 454. Una técnica parecida se utiliza en SNA para la autenticación mutua de dos LU cuando se vinculan para formar una sesión. La técnica se describe en "Autenticación a nivel de sesión" en la página 118.

Todas las técnicas anteriores para la autenticación mutua se pueden adaptar para proporcionar autenticación unidireccional.

Correlación de identidad en salidas de mensajes

Puede utilizar salidas de mensajes para procesar información para autenticar un ID de usuario, aunque puede ser mejor implementar la autenticación a nivel de aplicación.

Cuando una aplicación transfiere un mensaje a una cola, el campo *UserIdentifier* del descriptor de mensaje contiene un ID de usuario asociado a la aplicación. Sin embargo, no hay datos que se puedan utilizar para autenticar el ID de usuario. Estos datos se pueden añadir mediante una salida de mensajes en el extremo emisor de un canal y se pueden comprobar mediante una salida de mensajes en el extremo receptor del canal. Los datos de autenticación pueden ser una contraseña cifrada o una firma digital, por ejemplo.

Este servicio puede resultar más eficaz si se implementa a nivel de aplicación. El requisito básico es que el usuario de la aplicación que recibe el mensaje pueda identificar y autenticar al usuario de la aplicación que ha enviado el mensaje. Por lo tanto es natural considerar la implementación de este servicio a nivel de aplicación. Para obtener más información, consulte "Correlación de identidad en la salida de API y la salida cruzada de API" en la página 341.

Correlación de identidad en la salida de API y la salida cruzada de API

Una aplicación que recibe un mensaje debe poder identificar y autenticar al usuario de la aplicación que ha enviado el mensaje. Este servicio normalmente se implementa mejor a nivel de aplicación. Las salidas de API pueden implementar el servicio de varias maneras.

En cuanto a un mensaje individual se refiere, la identificación y autenticación son un servicio en el que participan dos usuarios, el emisor y el receptor del mensaje. El requisito básico es que el usuario de la aplicación que recibe el mensaje pueda identificar y autenticar al usuario de la aplicación que ha enviado el mensaje. Tenga en cuenta que el requisito de autenticación es unidireccional y no bidireccional.

Dependiendo de cómo se implemente, es posible que los usuarios y sus aplicaciones necesiten una interfaz o deban interactuar con el servicio. Además, cuándo y cómo se utiliza el servicio dependerá de dónde están ubicados los usuarios y sus aplicaciones y de la naturaleza de las aplicaciones propiamente dichas. Por lo tanto, es natural considerar la implementación del servicio a nivel de aplicación en lugar de a nivel de enlace.

Si piensa implementar este servicio a nivel de enlace, es posible que deba tener en cuenta algunos aspectos como, por ejemplo, los siguientes:

- Cómo aplicará el servicio, en un canal de mensajes, solamente a los mensajes que lo requieren
- Cómo permitirá que los usuarios y las aplicaciones se comuniquen o interactúen con el servicio, si éste es un requisito
- Dónde invocará los componentes del servicio en una situación de varios saltos, en la que se envía un mensaje a través de más de un canal hasta llegar a su destino

A continuación se muestran algunos ejemplos de cómo se puede implementar el servicio de identificación y autorización a nivel de aplicación. El término *salida de API* significa una salida de API o una salida cruzada de API.

- Cuando una aplicación transfiere un mensaje a una cola, una salida de API puede obtener una señal de autenticación de un servidor de autenticación fiable, como Kerberos. La salida de API puede añadir esta señal a los datos de aplicación del mensaje. Cuando la aplicación receptora recupera el mensaje, una segunda salida de API puede solicitar al servidor de autenticación que autentique al emisor comprobando la señal.
- Cuando una aplicación transfiere un mensaje a una cola, se puede añadir una salida de API a los elementos siguientes de los datos de aplicación del mensaje:
  - El certificado digital del emisor
  - La firma digital del emisor

  Si se dispone de algoritmos diferentes para generar un resumen del mensaje, la salida de API puede incluir el nombre del algoritmo que ha utilizado.

  Cuando la aplicación receptora recupera el mensaje, una segunda salida de API puede realizar las comprobaciones siguientes:
  - La salida de API puede validar el certificado digital analizando la cadena de certificados hasta llegar al certificado de la CA raíz. Para hacerlo, la salida de API debe tener acceso al repositorio de claves que contiene los certificados restantes de la cadena de certificados. Esta comprobación asegura que el emisor, identificado mediante el Nombre distinguido, es el propietario genuino de la clave pública que contiene el certificado.
  - La salida de API puede comprobar la firma digital utilizando la clave pública que contiene el certificado. Esta comprobación autentica al emisor.

  El Nombre distinguido del emisor se puede enviar en lugar del certificado digital completo. En este caso, el repositorio de claves debe contener el certificado del emisor, de modo que la segunda salida de API pueda buscar la clave pública del emisor. Otra posibilidad es enviar todos los certificados de la cadena de certificados.
- Cuando una aplicación transfiere un mensaje a una cola, el campo *UserIdentifier* del descriptor de mensaje contiene un ID de usuario asociado a la aplicación. El ID de usuario se puede utilizar para identificar al emisor. Para habilitar la autenticación, una salida de API puede añadir algunos datos como, por ejemplo, una contraseña cifrada, a los datos de la aplicación que contiene el mensaje. Cuando la aplicación receptora recupera el mensaje, una segunda rutina de API puede autenticar el ID de usuario utilizando los datos que ha transportado el mensaje.

  Esta técnica puede considerarse suficiente en los mensajes que se originan en un entorno controlado y fiable, y en aquellas circunstancias en las que no se disponga de un servidor de autenticación fiable o de soporte para PKI.

Trabajar con certificados revocados

Las Entidades emisoras de certificados pueden revocar los certificados digitales. Puede comprobar el estado de revocación de los certificados utilizando OCSP, o listas de revocación de certificados (CRL) en servidores LDAP, dependiendo de la plataforma.

Durante el reconocimiento TLS, los participantes en la comunicación se autentican entre sí mediante certificados digitales. La autenticación puede incluir una comprobación de que el certificado recibido continúa siendo fiable. Las Entidades emisoras de certificados (CA) revocan certificados por diversas razones, entre ellas:

- El propietario ha cambiado de organización
- La clave privada ya no es secreta

Las CA publican los certificados personales revocados en una Lista de revocación de certificados (CRL). Los certificados de CA que se han revocado se publican en una Lista de revocación de autorizaciones (ARL).

En las plataformas siguientes, el soporte de IBM MQ SSL comprueba si hay certificados revocados utilizando OCSP (Online Certificate Status Protocol) o utilizando CRL y ARL en servidores LDAP (Lightweight Directory Access Protocol). El OCSP es el método preferido.

- 

Linux
- 

UNIX
- 

Windows

IBM MQ classes for Java y IBM MQ classes for JMS no pueden utilizar la información de OCSP en un archivo de tabla de definiciones de canal de cliente. Sin embargo, puede configurar OCSP tal como se describe en Utilización de Online Certificate Protocol.

En las plataformas siguientes, y el soporte de IBM MQ SSL comprueba si hay certificados revocados utilizando CRL y ARL solo en servidores LDAP.

- 

IBM i
- 

z/OS

Para obtener más información sobre las Entidades emisoras de certificados, consulte “Certificados digitales” en la página 10.

Comprobación OCSP/CRL

La comprobación del protocolo de estado de certificados en línea (OCSP) /Lista de revocación de certificados (CRL) se realiza con respecto a los certificados entrantes remotos. El proceso comprueba toda la cadena implicada desde el certificado personal del sistema remoto hasta su certificado raíz.

Utilización de openSSL para verificar la validación de OCSP

Si la empresa utiliza openSSL para validar OCSP y, a continuación, intenta utilizar una conexión TLS de GSKit, recibirá un aviso de estado UNKNOWN.

Esto se debe a que GSKit comprueba el estado de revocación de todos los certificados de la cadena, aparte de la raíz. La operación de GSKit se ajusta a RFC 5280 y esto se describe en la política de confianza de GSKit. El algoritmo GSKit intenta todos los orígenes disponibles para la información de revocación, tal como se describe en RFC 5280 y la política de confianza de GSKit.

¿Cómo funciona la comprobación OCSP/CRL en IBM MQ?

IBM MQ da soporte a dos mecanismos para controlar el comportamiento al comprobar certificados en puntos finales OCSP o CRL con nombre, ya sea en la extensión de certificado o, tal como se define en los objetos AUTHINFO:

- Los atributos **OCSPCheckExtensions**, **CDPCheckExtensions**y **OCSPAuthentication** de la stanza SSL de del archivo qm.iniy
- Utilizando el parámetro SSLCRLNL del gestor de colas y las configuraciones AUTHINFO OCSP y CRLLDAP. Consulte ALTER AUTHINFO y ALTER QMGR para obtener más información.

Atención:

El mandato ALTER AUTHINFO con **AUTHTYPE(OCSP)** no se aplica para su uso en gestores de colas IBM i o z/OS . Sin embargo, se puede especificar en esas plataformas para copiarlas en la tabla de definición de canal de cliente (CCDT) para su uso por parte del cliente.

Los atributos de stanza SSL **OCSPCheckExtensions** y **CDPCheckExtensions** controlan si IBM MQ verificará un certificado en el servidor OCSP o CRL detallado dentro de la extensión AIA del certificado.

Si no está habilitado, no se contacta con el servidor OCSP o CRL de la extensión de certificado.

Si los servidores OCSP o CRL se detallan a través de objetos AUTHINFO y se hace referencia a ellos utilizando el atributo SSLCRLNL **QMGR** , durante el proceso de revocación de certificados, IBM MQ intenta ponerse en contacto con estos servidores.

Importante: Sólo se puede definir un objeto OCSP AUTHINFO en la lista de nombres SSLCRLNL.

If:

OCSPCheckExtensions= NO y **CDPCheckExtensions**=*NO* están establecidos, y
No hay servidores OCSP o CRL definidos en objetos AUTHINFO

no se realiza ninguna comprobación de revocación de certificados.

Al verificar un certificado para su estado de revocación, IBM MQ se pone en contacto con los servidores OCSP o CRL nombrados en el orden siguiente, si está habilitado:

1. El servidor OCSP detallado en un objeto **AUTHTYPE(OCSP)** y al que se hace referencia en el atributo SSLCRLNL **QMGR** .
2. Servidores OCSP detallados en la extensión AIA de los certificados, si **OCSPCheckExtensions**=*YES*.
3. Servidores CRL detallados en la extensión **CRLDistributionPoints** de los certificados, si **CDPCheckExtensions** =*YES*.
4. Cualquier servidor CRL detallado en objetos **AUTHINFO(CRLLDAP)** y al que se hace referencia en el atributo SSLCRLNL **QMGR** .

Al verificar un certificado, si un paso hace que el servidor OCSP o el servidor CRL devuelvan una respuesta REVOKED o VALID definitiva a una consulta para el certificado, no se realizan más comprobaciones y el estado del certificado tal como se presenta se utiliza para determinar si se debe confiar en él o no.

Si un servidor OCSP o un servidor CRL devuelve un resultado de UNKNOWN, el proceso continúa hasta que un servidor OCSP o CRL devuelve un resultado definitivo o hasta que se agotan todas las opciones.

El comportamiento de si un certificado se considera revocado, si no se puede determinar su estado, es diferente para los servidores OCSP y CRL:

- Para servidores CRL, si no se puede obtener ninguna CRL, el certificado se considera NOT_REVOKED
- Para servidores OCSP, si no se puede obtener ningún estado de revocación de un servidor OCSP con nombre, el comportamiento se controla mediante el atributo **OCSPAuthentication** en la stanza SSL del archivo qm.ini .

  Puede configurar este atributo para bloquear una conexión, permitir una conexión o permitir una conexión con un mensaje de aviso.

Puede utilizar el atributo **SSLHTTPProxyName**=*string* en la stanza SSL de los archivos qm.ini y mqclient.ini para las comprobaciones OCSP si es necesario. La serie es el nombre de host o la dirección de red del servidor proxy HTTP que GSKit va a utilizar para las comprobaciones de OCSP.

A partir de IBM MQ 9.1.5 , puede establecer el valor **OCSPTimeout** en la stanza SSL de los archivos qm.ini o mqclient.ini que establece el número de segundos que se debe esperar a que un programa de respuesta OCSP realice una comprobación de revocación.

ULW Certificados revocados y OCSP

IBM MQ determina qué programa de respuesta OSCP (Online Certificate Status Protocol) se utilizará y gestiona la respuesta recibida. Puede que tenga realizar pasos para que el canal de respuesta OCSP sea accesible.

Nota: Esta información solo se aplica a IBM MQ en los sistemas UNIX, Linux, and Windows.

Para comprobar el estado de revocación de un certificado digital utilizando OCSP, IBM MQ puede utilizar dos métodos para determinar el programa de respuesta OCSP con el que contactar:

- Utilizar la extensión de certificado AuthorityInfoAccess (AIA) en el certificado que se va a comprobar.
- Utilizar un URL especificado en un objeto de información de autenticación o especificado por una aplicación cliente.

Un URL especificado en un objeto de información de autenticación o mediante una aplicación cliente tiene prioridad sobre un URL en una extensión de certificados AIA.

Si el URL del programa de respuesta OCSP se oculta detrás de un cortafuegos, vuelva a configurar el cortafuegos de modo que pueda accederse al programa de respuesta OCSP o configure un servidor proxy

OCSP. Especifique el nombre del servidor proxy utilizando la variable SSLHTTPProxyName en la stanza SSL. En sistemas cliente, también puede especificar el nombre del servidor proxy utilizando la variable de entorno MQSSLPROXY. Para obtener más detalles consulte la información relacionada.

Si no está preocupado si se revocan los certificados TLS, quizá porque está realizando la ejecución en un entorno de prueba, puede establecer OCSPCheckExtensions en NO en la stanza de SSL. Si establece esta variable, se hace caso omiso de la extensión de certificados AIA. No es probable que esta solución se pueda aceptar en un entorno de producción, donde probablemente no desea permitir el acceso de los usuarios que presentan certificados revocados.

La llamada para acceder al programa de respuesta OCSP puede generar uno de estos tres resultados:

Aceptable
El certificado es válido.

Revocado
El certificado se revoca.

Desconocido
Esta salida se puede deber a una de las tres razones siguientes:

- IBM MQ no puede acceder al programa de respuesta OCSP.
- El programa de respuesta OCSP ha enviado una respuesta, pero IBM MQ no puede verificar la firma digital de la respuesta.
- El programa de respuesta OCSP ha enviado una respuesta que indica que no hay datos de revocación para el certificado.

Si IBM MQ recibe una salida OCSP `Desconocido`, su comportamiento depende del valor del atributo OCSPAuthentication. Para gestores de colas, este atributo se conserva en una de las ubicaciones siguientes:

- Linux UNIX En la stanza SSL del archivo `qm.ini` en UNIX and Linux.
- Windows En el registro de Windows.

Este atributo se puede establecer utilizando IBM MQ Explorer. Para clientes, el atributo se conserva en la stanza SSL del archivo de configuración de cliente.

Si se recibe una salida `Desconocido` y OCSPAuthentication está establecido en REQUIRED (el valor predeterminado), IBM MQ rechaza la conexión y emite un mensaje de error del tipo AMQ9716. Si están habilitados mensajes de sucesos SSL del gestor de colas, se genera un mensaje de suceso SSL del tipo MQRC_CHANNEL_SSL_ERROR con ReasonQualifier establecido en MQRQ_SSL_HANDSHAKE_ERROR.

Si se recibe una salida `Desconocido` y OCSPAuthentication está establecido en OPTIONAL, IBM MQ permite que se inicie el canal SSL y no se genere ningún aviso o mensajes de suceso SSL.

Si se recibe una salida `Desconocido` y OCSPAuthentication está establecido en WARN, se inicia el canal SSL pero IBM MQ emite un mensaje de aviso del tipo AMQ9717 en el registro de errores. Si están habilitados los mensajes de sucesos SSL, se genera un mensaje de sucesos SSL del tipo MQRC_CHANNEL_SSL_WARNING con ReasonQualifier establecido en MQRQ_SSL_UNKNOWN_REVOCATION.

Firma digital de respuestas OCSP

Un programa de respuesta OCSP puede firmar sus respuestas de una de tres formas. El programa de respuesta le informará del método que se utiliza.

- El programa de respuesta OCSP puede firmarse digitalmente utilizando el mismo certificado CA que emitió el certificado que está comprobando. En este caso, no es necesario que configure ningún certificado adicional; los pasos que haya realizado para establecer la conectividad TLS serán suficientes para verificar la respuesta OCSP.
- La respuesta OCSP se puede firmar de forma digital utilizando otro certificado firmado por la misma entidad emisora de certificados (CA) que emitió el certificado que se está comprobando. El certificado

de firma se envía junto con la respuesta OCSP en este caso. El certificado transmitido del programa de respuesta OCSP debe tener una extensión de uso de claves ampliado establecida en `id-kp-OCSPSigning` para que sea fiable para este fin. Debido a que la respuesta OCSP se envía con el certificado que la firmó (y dicho certificado está firmado por una CA que ya es fiable para la conectividad TLS) no es necesaria ninguna configuración adicional del certificado.

- La respuesta OCSP se puede firmar digitalmente utilizando otro certificado que no esté relacionado directamente con el certificado que está comprobando. En este caso, la respuesta OCSP está firmada por un certificado emitido por el propio programa de respuesta OCSP. Debe añadir una copia del certificado del respondedor OCSP a la base de datos de claves del cliente o gestor de colas que lleva a cabo la comprobación OCSP; consulte “Adición de un certificado de CA o la parte pública de un certificado autofirmado a un repositorio de claves en UNIX, Linux, and Windows” en la página 309 . Cuando se añade un certificado CA, de forma predeterminada se añade como raíz fiable, que es el valor necesario en este contexto. Si este certificado no se añade, IBM MQ no puede verificar la firma digital en la respuesta de OCSP y la comprobación de OCSP tiene como resultado una salida Desconocido, lo que podría hacer que IBM MQ cerrara el canal, en función del valor de OCSPAuthentication.

OCSP (Online Certificate Status Protocol) en aplicaciones cliente de Java y JMS.

Debido a una limitación de la API de Java, IBM MQ puede utilizar la comprobación de revocación de certificados OCSP (Online Certificate Status Protocol) para sockets seguros TLS únicamente cuando se habilita OCSP para todo el proceso de la máquina virtual Java (JVM). Hay dos modos de habilitar OCSP para todos los sockets seguros de la JVM:

- Editar el archivo JRE java.security para incluir los valores de configuración de OCSP que se muestran en la Tabla 1 y reiniciar la aplicación.
- Utilizar la API java.security.Security.setProperty(), sujeta a cualquier política de Java Security Manager que esté en vigor.

Como mínimo, debe especificar uno de los valores ocsp.enable y ocsp.responderURL.

Nombre de propiedad	Descripción
ocsp.enable	El valor de esta propiedad es `true` o `false`. Si es `true`, se habilita la comprobación OCSP cuando se lleva a cabo la comprobación de revocación de certificados. Si el valor es `false` no está establecido, la comprobación OCSP está inhabilitada.
ocsp.responderURL	El valor de esta propiedad es un URL que identifica la ubicación del programa de respuesta OCSP. El siguiente es un ejemplo: `ocsp.responderURL=http://ocsp.example.net:80`. De forma predeterminada, la ubicación del programa de respuesta OCSP se determina de forma implícita a partir del certificado que se está validando. La propiedad se utiliza cuando en el certificado falta la extensión de Authority Information Access (definida en RFC 3280) o cuando requiere una alteración temporal.
ocsp.responderCertSubjectName	El valor de esta propiedad es el nombre del asunto del certificado del programa de respuesta OCSP. El siguiente es un ejemplo: `ocsp.responderCertSubjectName="CN=OCSP Responder, O=XYZ Corp"`. De forma predeterminada, el certificado del programa de respuesta OSCP es el del emisor del certificado que se está validando. Esta propiedad identifica el certificado del programa de respuesta OSCP cuando el valor predeterminado no se aplica. Su valor es una serie de nombre distinguido (definido en RFC 2253) que identifica un certificado en el conjunto de certificados que se suministran durante la validación de la vía de acceso de certificado. En los casos en los que el nombre del asunto no es suficiente para identificar de forma exclusiva el certificado, en su lugar, se deben utilizar las dos propiedades

Nombre de propiedad	Descripción
	ocsp.responderCertIssuerName y ocsp.responderCertSerialNumber. Cuando se establece esta propiedad, se omiten las propiedades ocsp.responderCertIssuerName y ocsp.responderCertSerialNumber.
ocsp.responderCertIssuerName	El valor de esta propiedad es el nombre del emisor del certificado del programa de respuesta OCSP. El siguiente es un ejemplo: `ocsp.responderCertIssuerName="CN=Enterprise CA, O=XYZ Corp"`. De forma predeterminada, el certificado del programa de respuesta OSCP es el del emisor del certificado que se está validando. Esta propiedad identifica el certificado del programa de respuesta OSCP cuando el valor predeterminado no se aplica. Su valor es una serie de nombre distinguido (definido en RFC 2253) que identifica un certificado en el conjunto de certificados que se suministran durante la validación de la vía de acceso de certificado. Cuando se establece esta propiedad, también se debe establecer la propiedad ocsp.responderCertSerialNumber. Esta propiedad se omite cuando se establece la propiedad ocsp.responderCertSubjectName.
ocsp.responderCertSerialNumber	El valor de esta propiedad es el número de serie del certificado del programa de respuesta OCSP. El siguiente es un ejemplo: `ocsp.responderCertSerialNumber=2A:FF:00`. De forma predeterminada, el certificado del programa de respuesta OSCP es el del emisor del certificado que se está validando. Esta propiedad identifica el certificado del programa de respuesta OSCP cuando el valor predeterminado no se aplica. Este valor es una serie de dígitos hexadecimales (pueden haber separadores de espacio o de signo de dos puntos) que identifica un certificado en el conjunto de certificados que se suministran durante la validación de la vía de acceso de certificado. Cuando se establece esta propiedad, también se debe establecer la propiedad ocsp.responderCertIssuerName. Esta propiedad se omite cuando se establece la propiedad ocsp.responderCertSubjectName.

Antes de habilitar OCSP de este modo, existen varios puntos a tener en cuenta:

- Cuando se establece configuración de OCSP, todos los sockets seguros del proceso de la JVM resultan afectados. En algunos casos, es posible que esta configuración tenga efectos colaterales no deseados cuando la JVM se comparte con otro código de la aplicación que utiliza los sockets seguros TLS. Asegúrese de que la configuración OCSP elegida sea adecuada para todas las aplicaciones que se ejecutan en la misma JVM.
- Cuando se aplica el mantenimiento a JRE es posible que se sobrescriba el archivo java.security. Preste atención cuando aplique el mantenimiento del producto y los arreglos temporales de Java para no sobrescribir el archivo java.security. Es posible que sea necesario volver a aplicar los cambios de java.security después de aplicar el mantenimiento. Por este motivo, en su lugar, puede definir la configuración de OCSP mediante la API java.security.Security.setProperty().
- Cuando se habilita la comprobación OCSP, ésta solo tiene efecto si también está habilitada la comprobación de revocación. La comprobación de revocación se habilita mediante el método `PKIXParameters.setRevocationEnabled()`.
- Si está utilizando el interceptor AMS de Java que se describe en la sección Habilitación de la comprobación OCSP en los interceptores, preste atención y evite utilizar una configuración de java.security de OCSP que entre en conflicto con la configuración OCSP de AMS en el archivo de configuración del almacén de claves.

Trabajar con listas de revocación de certificados y listas de revocación de autorizaciones

El soporte de IBM MQ para las CRL y las ARL varía según la plataforma.

El soporte de CRL y ARL en cada plataforma es el siguiente:

- En z/OS, SSL del sistema da soporte a las CRL y las ARL almacenadas en servidores LDAP por el producto Tivoli Public Key Infrastructure.
- En otras plataformas, el soporte de CRL y ARL cumple con las recomendaciones de perfil PKIX X.509 V2 CRL.

IBM MQ mantiene una memoria caché de las CRL y las ARL a las que se ha accedido en las últimas 12 horas.

Cuando un gestor de colas o un cliente IBM MQ MQI client recibe un certificado, comprueba la CRL para confirmar que el certificado sigue siendo válido. IBM MQ comprueba en primer lugar la memoria caché, si ésta existe. Si la CRL no está en la memoria caché, IBM MQ interroga a las ubicaciones del servidor CRL de LDAP en el orden en que aparecen en la lista de nombres de los objetos de información de autenticación especificados por el atributo *SSLCRLNL* , hasta que IBM MQ encuentre una CRL disponible. Si no se especifica la lista de nombres o si se especifica con un valor en blanco, las CRL no se comprueban.

Configuración de los servidores LDAP

Configure la estructura de Árbol de información de directorios de LDAP para que refleje la jerarquía de Nombres distinguidos de las CA. Para ello, utilice archivos de Formato de intercambio de datos LDAP (LDIF).

Configure la estructura de Árbol de información de directorios (DIT) de LDAP, de modo que utilice la jerarquía correspondiente a los nombres distinguidos de las CA que emiten los certificados y las CRL. Puede configurar la estructura DIT con un archivo que utilice el Formato de intercambio de datos LDAP (LDIF). También puede utilizar archivos LDIF para actualizar un directorio.

Los archivos LDIF son archivos de texto ASCII que contienen la información necesaria para definir objetos en un directorio LDAP. Los archivos LDIF contienen una o varias entradas, cada una de las cuales consta de un Nombre distinguido, como mínimo una definición de clase de objeto y, opcionalmente, varias definiciones de atributo.

El atributo `certificateRevocationList;binary` contiene una lista, con formato binario, de los certificados de usuario revocados. El atributo `authorityRevocationList;binary` contiene una lista con formato binario de certificados de CA revocados. Para la utilización con IBM MQ TLS, los datos binarios para estos atributos deben cumplir con el formato DER (Definite Encoding Rules). Para obtener más información acerca de los archivos LDIF, consulte la documentación que se proporciona con el servidor LDAP.

Figura 20 en la página 349 muestra un archivo LDIF de ejemplo que puede crear como entrada al servidor LDAP para cargar los CRL y ARL emitidos por CA1, que es una entidad emisora de certificados imaginaria con el nombre distinguido "CN=CA1, OU=Test, O=IBM, C=GB", configurado por la organización de prueba en 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 20. Archivo LDIF de ejemplo para una Entidad emisora de certificados. Puede variar de implementación en implementación.

La Figura 21 en la página 349 muestra la estructura DIT que el servidor LDAP crea cuando carga el archivo LDIF de ejemplo que se muestra en la Figura 20 en la página 349 junto con un archivo similar para la CA2, una Entidad emisora de certificados ficticia establecida por la organización PKI, también dentro de IBM.

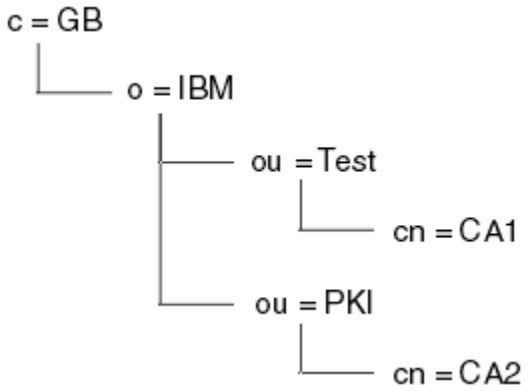


Figura 21. Ejemplo de una estructura de árbol de la información de directorios LDAP

WebSphere comprueba las CRL y las ARL.

Nota: Asegúrese de que la lista de control de accesos de su servidor LDAP permita que los usuarios autorizados lean, busquen y comparen las entradas que contienen las CRL y las ARL. WebSphere MQ accede al servidor LDAP utilizando las propiedades LDAPUSER y LDAPPWD del objeto AUTHINFO.

Configuración y actualización de los servidores LDAP
Utilice este procedimiento para configurar o actualizar el servidor LDAP.

1. Obtenga las CRL y ARL en formato DER de su autoridad o autoridades de certificación.
2. Con un editor de texto o la herramienta que le proporcione el servidor LDAP, cree uno o varios archivos LDIF que contengan el nombre distinguido de la CA y las definiciones de clases de objetos necesarias. Copie los datos con formato DER en el archivo LDIF como valores del atributo `certificateRevocationList;binary` para las CRL, del atributo `authorityRevocationList;binary` para las ARL, o ambos.
3. Inicie el servidor LDAP.
4. Añada las entradas del archivo o archivos LDIF que ha creado en el paso “2” en la página 349.

Cuando haya configurado el servidor CRL LDAP, compruebe que se ha configurado correctamente. Primero, intente utilizar un certificado que no se haya revocado en el canal, y compruebe que el canal

se inicia correctamente. A continuación, utilice un certificado que se haya revocado y compruebe que el canal no se inicia correctamente.

Obtenga las CRL actualizadas de las Autoridades de certificación de forma regular. Se recomienda que lo haga en sus servidores LDAP cada 12 horas.

Acceso a las CRL y las ARL con un gestor de colas

Un gestor de colas está asociado a uno o más objetos de información de autenticación, que contienen la dirección de un servidor CRL LDAP. IBM i IBM MQ en IBM i se comporta de forma diferente a otras plataformas.

Tenga en cuenta que en este apartado, la información sobre las listas de revocación de certificados (CRL) también es aplicable para las listas de revocación de autorizaciones (ARL).

Debe indicar al gestor de colas cómo acceder a las CRL proporcionándole objetos de información de autenticación, cada uno de los cuales contiene la dirección de un servidor CRL LDAP. Los objetos de información de autenticación se mantienen en una lista de nombres, que se especifica en el atributo de gestor de colas *SSLCRLNL* .

En el ejemplo siguiente, MQSC se utiliza para especificar los parámetros:

1. Defina los objetos de información de autenticación con el mandato MQSC, DEFINE AUTHINFO, con el parámetro AUTHTYPE establecido en CRLLDAP. IBM i En IBM i, también puede utilizar el mandato CL CRTMQMAUTI.

   El valor CRLLDAP para el parámetro AUTHTYPE indica que se accede a las CRL en servidores LDAP. Cada objeto de información de autenticación con el tipo CRLLDAP que cree contendrá la dirección de un servidor LDAP. Cuando tenga más de un objeto de información de autenticación, los servidores LDAP a los que apuntan deben contener información idéntica. Esto permite que el servicio continúe si uno o varios servidores LDAP no se ejecutan correctamente.

   z/OS Asimismo, únicamente en z/OS, se debe acceder a todos los servidores LDAP utilizando el mismo ID de usuario y contraseña. El ID de usuario y la contraseña utilizados son los especificados en el primer objeto AUTHINFO de la lista de nombres.

   En todas las plataformas, el ID de usuario y la contraseña se envían al servidor LDAP sin cifrar.
2. Con el mandato MQSC, DEFINE NAMELIST, defina una lista de nombres para los nombres de los objetos de información de autenticación. z/OS En z/OS, asegúrese de que el atributo de lista de nombres NLTYPE esté establecido en AUTHINFO.
3. Con el mandato MQSC, ALTER QMGR, proporcione la lista de nombres al gestor de colas. Por ejemplo:

   ```
   ALTER QMGR SSLCRLNL(sslcrlnlname)
   ```

   donde `sslcrlnlname` es la lista de nombres de los objetos de información de autenticación.

   Este mandato establece un atributo de gestor de colas denominado *SSLCRLNL*. El valor inicial del gestor de colas para este atributo está en blanco.

IBM i En IBM i, puede especificar objetos de información de autenticación, pero el gestor de colas no utiliza ni objetos de información de autenticación ni una lista de nombres de objetos de información de autenticación. Solamente los clientes IBM MQ que utilizan una tabla de conexiones de cliente generada por un gestor de colas de IBM i utilizan la información de autenticación especificada para dicho gestor de colas de IBM i. El atributo de gestor de colas *SSLCRLNL* en IBM i determina qué información de autenticación utilizan estos clientes. Consulte “Acceso a las CRL y las ARL en IBM i” en la página 351 para obtener información sobre cómo indicar a un gestor de colas de IBM i cómo acceder a las CRL.

Puede añadir hasta 10 conexiones a servidores LDAP alternativos a la lista de nombres, para asegurarse de la continuidad del servicio si uno o varios servidores LDAP no respondieran. Tenga en cuenta que los servidores LDAP deben contener información idéntica.

IBM i *Acceso a las CRL y las ARL en IBM i*

Utilice este procedimiento para acceder a las CRL o las ARL en IBM i.

Tenga en cuenta que en este apartado, la información sobre las listas de revocación de certificados (CRL) también es aplicable para las listas de revocación de autorizaciones (ARL).

Siga estos pasos para configurar una ubicación CRL para un certificado específico en IBM i:

1. Acceda a la interfaz DCM, tal y como se describe en el apartado "Acceso a DCM" en la página 278.
2. En la categoría de tareas **Gestionar ubicaciones CRL** del panel de navegación, pulse **Añadir ubicación CRL**. Se visualiza la página Ubicaciones CRL en la sección de tareas.
3. En el campo **Nombre de ubicación de CRL** , escriba un nombre de ubicación de CRL, por ejemplo `LDAP Server #1`
4. En el campo **Servidor LDAP**, escriba el nombre de servidor LDAP.
5. En el campo **Utilizar Secure Sockets Layer (SSL)**, seleccione **Sí** si desea conectarse al servidor LDAP utilizando TLS. De lo contrario, seleccione **No**.
6. En el campo **Número de puerto**, escriba un número de puerto para el servidor LDAP; por ejemplo, 389.
7. Si el servidor LDAP no permite que los usuarios anónimos consulte el directorio, escriba un nombre distinguido de inicio de sesión para el servidor en el campo **nombre distinguido de inicio de sesión**.
8. Pulse **Aceptar**. DCM le informa de que ha creado la ubicación CRL.
9. En el panel de navegación, pulse **Seleccionar un almacén de certificados**. La página Seleccionar un almacén de certificados se visualiza en la sección de tareas.
10. Seleccione el recuadro **Otro almacén de certificados del sistema** y pulse **Continuar**. Se visualiza la página Almacén de certificados y contraseña.
11. En el campo **Vía de acceso y nombre de archivo del almacén de certificados**, escriba la vía de acceso y nombre de archivo de IFS que estableció en el apartado "Crear un almacén de certificados en IBM i" en la página 279.
12. Escriba una contraseña en el campo **Contraseña del almacén de certificados**. Pulse **Continuar**. La página Almacén de certificados actual se visualiza en la sección de tareas.
13. En la categoría de tareas **Gestionar certificados** del panel de navegación, pulse **Actualizar asignación de ubicación CRL**. La página Asignación de ubicación CRL se visualiza en la sección de tareas.
14. Seleccione el botón correspondiente al certificado de CA al que desea asignar la ubicación CRL. Pulse **Actualizar asignación de ubicación CRL**. La página Actualizar asignación de ubicación CRL se visualiza en la sección de tareas.
15. Seleccione el botón para la ubicación CRL que desea asignar al certificado. Pulse **Actualizar asignación**. DCM le informa de que ha actualizado la asignación.

Tenga en cuenta que DCM le permite asignar un servidor LDAP diferente mediante la Entidad emisora de certificados.

Acceso a las CRL y las ARL utilizando IBM MQ Explorer

Puede utilizar IBM MQ Explorer para indicar a un gestor de colas cómo acceder a las CRL.

Tenga en cuenta que en este apartado, la información sobre las listas de revocación de certificados (CRL) también es aplicable para las listas de revocación de autorizaciones (ARL).

Utilice el procedimiento siguiente para establecer una conexión LDAP con una CRL:

1. Asegúrese de que ha iniciado el gestor de colas.
2. Pulse el botón derecho del ratón en la carpeta **Información de autenticación** y pulse **Nuevo -> Información de autenticación**. En la hoja de propiedades que se abrirá:
   a. En la primera página **Crear información de autenticación**, escriba un nombre para el objeto CRL(LDAP).

b. En la página **General** de **Modificar las propiedades**, seleccione el tipo de conexión. De manera opcional, puede escribir una descripción.

c. Seleccione la página **CRL(LDAP)** de **Modificar las propiedades**.

d. Escriba el nombre del servidor LDAP como el nombre de red o la dirección IP.

e. Si el servidor requiere detalles para la conexión, proporcione un ID de usuario y, si es necesario, una contraseña.

f. Pulse **Aceptar**.

3. Pulse el botón derecho del ratón en la carpeta Listas de nombres y pulse **Nuevo-> Lista de nombres**. En la hoja de propiedades que se abrirá:

a. Escriba un nombre para la lista de nombres.

b. Añada a la lista el nombre del objeto CRL(LDAP) (del paso "2.a" en la página 351).

c. Pulse **Aceptar**.

4. Pulse el botón derecho del ratón en el gestor de colas, seleccione **Propiedades** y seleccione la página **SSL**:

a. Seleccione el recuadro de selección **Comprobar los certificados enviados a este gestor de colas respecto a las listas de revocación de certificados**.

b. Escriba el nombre de la lista de nombres (del paso "3.a" en la página 352) en el campo **Nombre de la lista de CRL**.

Acceso a las CRL y las ARL con un IBM MQ MQI client

Existen tres formas de especificar los servidores LDAP que contienen listas de revocación de certificados (CRL) para su comprobación por parte de un IBM MQ MQI client.

Tenga en cuenta que en este apartado, la información sobre las listas de revocación de certificados (CRL) también es aplicable para las listas de revocación de autorizaciones (ARL).

A continuación se indican las tres formas de especificar los servidores LDAP:

- Utilizar una tabla de definiciones de canal
- Utilizar la estructura de opciones de configuración de SSL, MQSCO, en una llamada MQCONNX
- Utilizar Active Directory (en sistemas Windows con soporte para Active Directory)

Para más detalles, consulte la información relacionada.

Puede incluir hasta 10 conexiones a servidores LDAP alternativos para asegurarse de la continuidad del servicio si uno o más servidores LDAP no se realiza correctamente n. Tenga en cuenta que los servidores LDAP deben contener información idéntica.

No puede acceder a las CRL de LDAP desde un canal de cliente MQI de IBM MQ MQI client que se ejecuta en Linux (plataforma zSeries).

Ubicación de un programa de respuestas OCSP, y de servidores LDAP que contienen CRL

En un sistema de IBM MQ MQI client, puede especificar la ubicación de un programa de respuestas OCSP y de servidores LDAP (Lightweight Directory Access Protocol) que tienen las CRL (listas de revocación de certificados).

Puede especificar estas ubicaciones de tres formas, se describen aquí en orden de mayor a menor prioridad.

IBM i Para IBM i, consulte Acceso a las CRL y las ARL en IBM i.

Cuando una aplicación cliente IBM MQ MQI client emite una llamada MQCONNX

Puede especificar una respuesta OCSP o un servidor LDAP que contiene CRL en una llamada **MQCONNX**.

En una llamada **MQCONNX**, la estructura de opciones de conexión, MQCNO, puede hacer referencia a una estructura de opciones de configuración SSL, MQSCO. A su vez, la estructura MQSCO puede hacer

referencia a una o varias estructuras de registro de información de autenticación, MQAIR. Cada estructura MQAIR contiene toda la información que un IBM MQ MQI client necesita para acceder a una respuesta OCSP o un servidor LDAP que contiene CRL. Por ejemplo, uno de los campos de una estructura MQAIR es el URL en el que se puede contactar con una respuesta. Para obtener más información acerca de la estructura MQAIR, consulte MQAIR - Registro de información de autenticación.

Utilización de una tabla de definiciones de canal de cliente (CCDT) para acceder a un programa de respuestas OCSP o a servidores LDAP

Para que un IBM MQ MQI client pueda acceder a un programa de respuestas OCSP o a servidores LDAP que contienen CRL, incluya los atributos de uno o más objetos de información de autenticación en una tabla de definiciones de canal de cliente.

En un gestor de colas de servidor, puede definir uno o varios objetos de información de autenticación. Los atributos de un objeto de autenticación contienen toda la información necesaria para acceder a un programa de respuestas OCSP (en plataformas donde se admite OCSP) o a un servidor LDAP que contiene CRL. Uno de los atributos especifica el URL del programa de respuestas OCSP, el otro especifica la dirección de host, o la dirección IP, de un sistema en que se ejecuta un servidor LDAP.

z/OS IBM i Un objeto de información de autenticación con AUTHTYPE(OCSP) no es aplicable para utilizarse en gestores de colas de IBM i o z/OS, pero puede especificarse en las plataformas que deben copiarse a la tabla de definiciones del canal de cliente (CCDT) para que las use el cliente.

Para que un cliente MQI de IBM MQ MQI client pueda acceder a un programa de respuestas OCSP o a servidores LDAP que contienen CRL, pueden incluirse los atributos de uno o más objetos de información de autenticación en una tabla de definiciones de canal de cliente. Puede incluir dichos atributos de una de las maneras siguientes:

Multi

En plataformas de servidor AIX, Linux, IBM i,Solaris y Windows

Puede definir una lista de nombres que contenga los nombres de uno o varios objetos de información de autenticación. A continuación, puede establecer el atributo del gestor de colas, **SSLCRLNL**, en el nombre de esta lista de nombres.

Si utiliza CRL, puede configurarse más de un servidor LDAP para ofrecer más disponibilidad. La intención es que cada servidor LDAP contenga las mismas CRL. Si un servidor LDAP no está disponible cuando se necesita, un IBM MQ MQI client puede intentar acceder a otro.

Los atributos de los objetos de información de autenticación identificados por la lista de nombres se denominan colectivamente *ubicación de la revocación del certificado*. Cuando establece el atributo de gestor de colas, **SSLCRLNL**, en el nombre de la lista de nombres, la ubicación de revocación de certificados se copia en la tabla de definiciones de canal de cliente asociada con el gestor de colas. Si puede accederse a la CCDT desde un sistema cliente como archivo compartido, o si la CCDT se copia posteriormente en un sistema de cliente, el IBM MQ MQI client de dicho sistema puede utilizar la ubicación de revocación de certificados de la CCDT para acceder a un programa de respuesta OCSP o a servidores LDAP que contienen CRL.

Si la ubicación de la revocación del certificado del gestor de colas se cambia posteriormente, el cambio se refleja en la CCDT asociada al gestor de colas. Si el atributo de gestor de colas, **SSLCRLNL**, se establece en blanco, la ubicación de revocación de certificado se elimina de la CCDT. Estos cambios no quedan reflejados en ninguna copia de la tabla en un sistema cliente.

Si necesita que la ubicación de revocación del certificado en los extremos del cliente y del servidor de un canal MQI sea diferente, y ha utilizado el gestor de colas del servidor para crear la información de la ubicación de revocación del certificado, puede hacer lo siguiente:

1. En el gestor de colas del servidor, cree la información de la ubicación de revocación del certificado que se utilizará en el sistema cliente.
2. Copie la CCDT que contiene la ubicación de revocación del certificado al sistema de cliente.

3. En el gestor de colas del servidor, cambie la ubicación de revocación del certificado por la que se necesita en el extremo del servidor del canal MQI.
4. En la máquina de cliente, puede utilizar el mandato **runmqsc** con el parámetro **-n**.

Multi

En plataformas cliente AIX, Linux, IBM i,Solaris y Windows

Puede crear una CCDT en la máquina cliente utilizando el mandato runmqsc con el parámetro **-n** y los objetos **DEFINE AUTHINFO** en el archivo CCDT. El orden en que se definen los objetos será el orden en que se utilizan en el archivo. Cualquier nombre que utilice en un objeto **DEFINE AUTHINFO** no se retendrá en el archivo. Solo se utilizan números de posición cuando ejecute **DISPLAY** para los objetos **AUTHINFO** de un archivo CDDT.

Nota: Si especifica el parámetro **-n**, no debe especificar ningún otro parámetro.

Utilización de Active Directory en Windows

Windows

En sistemas Windows, puede utilizar el mandato de control **setmqcrl** para publicar la información de CRL actual en Active Directory.

El mandato **setmqcrl** no publica información OCSP.

Para obtener información sobre este mandato y su sintaxis, consulte setmqcrl.

Acceso a las CRL y las ARL con un IBM MQ classes for Java y IBM MQ classes for JMS

IBM MQ classes for Java y IBM MQ classes for JMS acceden a las CRL de forma diferente de otras plataformas.

Para obtener información sobre cómo trabajar con CRL y ARL con IBM MQ classes for Java, consulte Utilización de listas de revocación de certificados

Si desea más información sobre cómo trabajar con CRL y ARL con IBM MQ classes for JMS, consulte Propiedad de objeto SSLCERTSTORES

Manipulación de objetos de información de autenticación

Puede manipular objetos de información de autenticación utilizando mandatos MQSC o PCF, o mediante IBM MQ Explorer.

Los mandatos MQSC siguientes actúan en los objetos de información de autenticación:

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

Si desea una descripción completa de estos mandatos, consulte Mandatos MQSC.

Los mandatos PCF (Programmable Command Format) siguientes actúan en los objetos de información de autenticación:

- Crear información de autenticación
- Copiar información de autenticación
- Modificar información de autorización
- Suprimir información de autenticación
- Consultar información de autenticación
- Consultar nombres de información de autenticación

Si desea una descripción completa de estos mandatos, consulte Definiciones de los formatos de mandato programables.

En plataformas donde esté disponible, también puede utilizar IBM MQ Explorer.

Linux UNIX Utilización de PAM (Pluggable Authentication Method)

Puede utilizar PAM únicamente en las plataformas UNIX and Linux. Un sistema UNIX típico tiene módulos PAM que implementan el mecanismo de autenticación tradicional; sin embargo, puede haber más. Del mismo modo que la tarea básica de validación de contraseñas, los módulos PAM también pueden invocarse para llevar a cabo reglas adicionales.

Los archivos de configuración definen qué método de autenticación se va a utilizar para cada aplicación. Las aplicaciones de ejemplo incluyen el inicio de sesión de terminal estándar, ftp y telnet.

La ventaja de PAM es que la aplicación no necesita saber, o preocuparse, de cómo se autentica en realidad el ID de usuario. Siempre que la aplicación pueda proporcionar una forma correcta de datos de autenticación para PAM, el mecanismo detrás es transparente.

El formato de los datos de autenticación depende del sistema que se utiliza. Por ejemplo, IBM MQ obtiene una contraseña a través de parámetros, como por ejemplo la estructura MQCSP utilizada en la llamada de API `MQCONNX`.

Importante: No puede establecer el atributo **AUTHENMD** hasta que instale IBM MQ 8.0.0 Fix Pack 3y, a continuación, reinicie el gestor de colas, utilizando un **-e CMDLEVEL=***nivel* de *802* (en el mandato strmqm) para establecer el nivel de mandatos que necesita.

Configuración del sistema para utilizar PAM

El nombre de servicio utilizado por IBM MQ, al invocar PAM, es *ibmmq*.

Tenga en cuenta que una instalación de IBM MQ intenta mantener una configuración PAM predeterminada que permite conexiones para los usuarios del sistema operativo, basándose en valores predeterminados conocidos para los distintos sistemas operativos.

Sin embargo, el administrador del sistema debe verificar qué reglas definidas en los archivos `/etc/pam.conf` o `/etc/pam.d/ibmmq` siguen siendo apropiadas.

Autorización del acceso a objetos

Esta sección contiene información sobre cómo utilizar el gestor de autorizaciones sobre objetos y programas de salida de canal para controlar el acceso a los objetos.

ULW En los sistemas UNIX, Linux, and Windows, el acceso a los objetos se controla utilizando el gestor de autorizaciones sobre objetos (OAM). Esta colección de temas contiene información sobre cómo utilizar la interfaz de mandatos en el OAM.

Esta sección también contiene una lista de comprobación que puede utilizar para determinar qué tareas realizar para aplicar seguridad al sistema en todas las plataformas, y las consideraciones para otorgar a usuarios la autorización para administrar IBM MQ y para trabajar con objetos IBM MQ.

Si los mecanismos de seguridad proporcionados no satisfacen sus necesidades, puede desarrollar sus propios programas de salida de canal.

Determinar qué usuario se utiliza para la autorización

Las autorizaciones para acceder a los recursos se otorgan a los grupos de los que el usuario es miembro o, en determinadas modalidades, directamente al usuario asociado a la conexión. Durante el proceso de conexión, y en particular para las conexiones remotas (cliente), la configuración del gestor de colas podría cambiar esta identidad. Esta página lista las distintas características de IBM MQ y sus opciones de configuración que podrían afectar a la identidad de una aplicación de conexión y al orden de prioridad en el que estas características entran en vigor.

Características que pueden modificar qué usuario se adopta

Las distintas características que pueden establecer qué usuario debe estar autorizado son las siguientes:

Usuario confirmado por aplicación
Cuando IBM MQinicia una conexión remota, el usuario del sistema operativo con el que se ejecuta el proceso se envía al gestor de colas receptor. Este usuario se envía para asegurarse de que si no existe ninguna configuración adicional que modifique el usuario, hay un usuario que se puede utilizar para la comprobación de autorización.

No se recomienda utilizar este usuario como base para la autorización, ya que permite a las conexiones confirmar su identidad sin ninguna validación del lado del servidor. Esto puede incluir incluso al usuario administrativo ('mqm ').

Valor MCAUSER de canal
Las aplicaciones que se conectan a través de enlaces de red lo hacen utilizando una definición de canal de IBM MQ . Las definiciones de canal dan soporte al atributo **MCAUSER** , que se puede utilizar para especificar un usuario diferente que se utilizará para la autorización en lugar del usuario confirmado por las aplicaciones de conexión.

Autenticación de conexión ADOPTCTX
Las aplicaciones pueden especificar un usuario y una contraseña para enviarlos a un gestor de colas con fines de autenticación. Estas credenciales se autentican utilizando la configuración especificada para la característica de autenticación de conexión. La opción **ADOPTCTX** para la autenticación de conexión controla si se debe utilizar un usuario para la autorización después de que se haya validado correctamente. Si se establece en YES, el usuario que se proporciona para la autenticación se adopta para las comprobaciones de autorización.

Registro de autenticación de canal MCAUSER
Durante el proceso de conexión, el gestor de colas intentará encontrar un registro de autenticación de canal que coincida con la conexión. Si un registro de autenticación de canal coincide y su valor de atributo **USERSRC** se establece en MAP, IBM MQ cambia el usuario utilizado para las autorizaciones al valor del atributo **MCAUSER** .

Salidas de seguridad
Las salidas de seguridad son funciones personalizadas que se pueden escribir y llamar durante el proceso de seguridad de IBM MQ . Cuando se llama a la función, se proporciona con una copia de la estructura MQCD que incluye varios campos relacionados con el usuario de conexiones que se utilizará para las comprobaciones de autorización. Las salidas de seguridad pueden modificar estos campos para cambiar el usuario que se autorizará.

orden de prioridad

La tabla siguiente muestra el orden de prioridad para cada característica de seguridad descrita en “Características que pueden modificar qué usuario se adopta” en la página 356 cuando IBM MQ selecciona un usuario para autorizar. El orden es de menor a mayor, es decir, una característica de seguridad que establece un usuario en la primera fila se altera temporalmente por cualquiera de las otras filas.

Tabla 68. Orden de prioridad para las características de seguridad

Orden	**Característica**
1 (más bajo)	ID confirmado de aplicación
2	Atributo **MCAUSER** de definición de canal
3	Autenticación de conexión con **ADOPTCTX(YES)**
4	Registros de autenticación de canal con **USERSRC(MAP)**
5 (más alto)	Salida de seguridad

Implicaciones de la adopción temprana

Los registros de autenticación de conexión y autenticación de canal proporcionan una opción de configuración que controla cuándo se realiza la adopción del usuario de autenticación de conexión. Este valor se conoce como adopción temprana. Si la adopción temprana está habilitada, la adopción de identidad de autenticación de conexión se produce antes de que se procesen los registros de autenticación de canal (lo que significa que los registros de autenticación de canal alteran temporalmente cualquier adopción de **CONNAUTH**).

Si está inhabilitado, el orden se invierte, es decir, los registros de autenticación de canal se procesan antes de la adopción de **CONNAUTH** . En esta situación, la adopción de autenticación de conexión tiene una prioridad efectiva más alta que los registros de autenticación de canal.

El valor predeterminado para la adopción temprana es `enabled`.

ULW Control del acceso a objetos mediante el OAM en UNIX, Linux, and Windows

El Gestor de autorizaciones sobre objetos (OAM) proporciona una interfaz de mandatos para otorgar y revocar autorización a objetos IBM MQ.

Debe tener la autorización adecuada para utilizar estos mandatos, como se describe en "Autorización para administrar IBM MQ en UNIX, Linux, and Windows" en la página 409. Los ID de usuario que están autorizados para administrar IBM MQ tiene autorización de *superusuario* para el gestor de colas, lo que significa que no tiene que otorgarles más permisos para emitir mandatos o solicitudes MQI.

Linux UNIX Permisos basados en usuario de OAM en UNIX and Linux

Desde IBM MQ 8.0, en sistemas UNIX and Linux, el gestor de autorizaciones sobre objetos (OAM) puede utilizar la autorización basada en usuario, así como la autorización basada en grupo.

Antes de IBM MQ 8.0, las listas de control de accesos (ACL) en UNIX and Linux solo se basan en grupos. A partir de IBM MQ 8.0, las ACL se basan tanto en ID de usuario como en grupos, y puede utilizar el modelo basado en usuario o el modelo basado en grupo para la autorización estableciendo el atributo **SecurityPolicy** en el valor apropiado, tal como se describe en Configuración de servicios instalables y Configuración de stanzas de servicio de autorización en UNIX y Linux.

Cambios en el comportamiento para IBM MQ 8.0 y posteriores

Desde IBM MQ 8.0, al ejecutarse con la política basada en usuario, algunos mandatos devuelven información diferente respecto a versiones anteriores del producto.

- Los mandatos **dmpmqaut** y **dmpmqcfg** muestran registros basados en usuario, ya que realizan las operaciones equivalentes de PCF.
- El plug-in OAM para IBM MQ Explorer muestra registros basados en usuario y permite modificaciones basadas en usuario.
- La función **Inquire** de OAM devuelve resultados que muestran que tiene capacidad de usuario.

La utilización del atributo **-p** en el mandato **setmqaut** no otorga acceso a todos los usuarios del mismo grupo primario, cuando las autorizaciones basadas en usuario están habilitadas en el archivo `qm.ini` tal como se describe en Stanza de servicio del archivo qm.ini.

Si comienza a utilizar la autorización basada en usuario y tiene muchos usuarios, probablemente, habrá más registros almacenados en la cola AUTH que con el modelo basado en grupo, y el proceso de autorización podría tardar un poco más que antes, ya que hay más registros para verificar. No se espera que este aumento sea significativo. Si es necesario, puede utilizar una combinación de permisos de usuario y grupo.

Consideraciones sobre la migración

Si cambia el modelo de grupo a usuario para un gestor de colas existente, no se produce ningún efecto inmediato. Las autorizaciones que ya se han realizado se siguen aplicando. Cualquier usuario que se conecta al gestor de colas recibe los mismos privilegios que antes: la combinación de todos los grupos a los que pertenece su ID. Cuando se emiten nuevos mandatos **setmqaut** para los ID de usuario, surten efecto de forma inmediata.

Si crea un gestor de colas nuevo con la política de usuario, este gestor de colas solo tiene permisos para el usuario que lo ha creado (que normalmente suele ser, aunque no necesariamente siempre, el ID de usuario `mqm`). También hay permisos que se otorgan automáticamente al grupo `mqm`. Sin embargo, si no tiene `mqm` como grupo principal, el grupo `mqm` no está incluido en el conjunto inicial de autorizaciones.

Si pasa de una política de usuario a una política de grupo, las autorizaciones basadas en usuario no se suprimen automáticamente. Sin embargo, dejan de utilizarse durante la comprobación de permisos. Antes de revertir la política, guarde la configuración actual, cambie la política y reinicie el gestor de colas y, después, reproduzca el script. Puesto que ahora es un gestor de colas basado en grupo, la consecuencia es que las reglas de ID de usuario se almacenan basándose en el grupo principal.

Conceptos relacionados

Gestor de autorizaciones sobre objetos (OAM)

Principales y grupos en UNIX, Linux y Windows

Stanza de servicio del archivo qm.ini

Referencia relacionada

Mandato **crtmqm** (crear gestor de colas)

ULW Otorgar acceso a un objeto IBM MQ en UNIX, Linux, and Windows

Utilice el mandato de control **setmqaut**, el mandato MQSC **SET AUTHREC** o el mandato PCF **MQCMD_SET_AUTH_REC** para otorgar a usuarios, y grupos de usuarios, acceso a los objetos IBM MQ. Tenga en cuenta que en IBM MQ Appliance solo puede utilizar el mandato **SET AUTHREC** .

Para obtener una definición completa del mandato de control **setmqaut** y su sintaxis, consulte setmqaut.

Para obtener una definición completa del mandato MQSC **SET AUTHREC** y su sintaxis, consulte SET AUTHREC.

Para obtener una definición completa del mandato PCF **MQCMD_SET_AUTH_REC** y su sintaxis, consulte Establecer registro de autorización.

El gestor de colas debe estar en ejecución para poder utilizar este mandato. Cuando haya modificado el acceso de un principal, el OAM reflejará inmediatamente los cambios.

Para otorgar a los usuarios acceso a un objeto, debe especificar:

- El nombre del gestor de colas que es el propietario de los objetos con los que está trabajando; si no especifica el nombre de un gestor de colas, se utilizará el gestor de colas predeterminado.
- El nombre y el tipo del objeto (para identificar el objeto de forma exclusiva). El nombre se especifica como un *perfil*; puede ser el nombre explícito del objeto o un nombre genérico que incluya caracteres comodín. Para obtener una descripción detallada de los perfiles genéricos y cómo se utilizan los caracteres comodín en los mismos, consulte el “Utilización de perfiles genéricos del OAM en UNIX, Linux, and Windows” en la página 360.
- Uno o varios principales y nombres de grupo a los que se aplica la autorización.

  Si un ID de usuario contiene espacios, póngalo entre signos de interrogación cuando utilice este mandato. En sistemas Windows, puede calificar un ID de usuario con un nombre de dominio. Si el ID de usuario real contiene un símbolo (@), sustitúyalo por @@ para mostrar que forma parte del ID de usuario, no el delimitador entre el ID de usuario y el nombre de dominio.
- Una lista de autorizaciones. Cada elemento de la lista especifica un tipo de acceso que se va a otorgar (o revocar) para este objeto. Cada autorización de la lista se especifica como una palabra clave, con un signo más (+) o un signo menos (-) como prefijo. Utilice un signo más para añadir la autorización

especificada y un signo menos para eliminar la autorización. No debe haber ningún espacio entre el signo + o - y la palabra clave.

Se puede especificar cualquier número de autorizaciones en un solo mandato. Por ejemplo, la lista de autorizaciones que permite que un usuario o un grupo transfiera los mensajes a una cola y los examine pero revoca el acceso para la obtención de mensajes es:

```
+browse -get +put
```

Ejemplos de cómo utilizar el mandato `setmqaut`

Los ejemplos siguientes muestran cómo utilizar el mandato `setmqaut` para otorgar y revocar el permiso para utilizar un objeto:

```
setmqaut -m saturn.queue.manager -t queue -n RED.LOCAL.QUEUE
         -g groupa +browse -get +put
```

En este ejemplo:

- `saturn.queue.manager` es el nombre del gestor de colas.
- queue es el tipo de objeto.
- `RED.LOCAL.QUEUE` es el nombre del objeto.
- `groupa` es el identificador del grupo cuyas autorizaciones se van a modificar.
- `+browse -get +put` es la lista de autorizaciones para la cola especificada.
  - `+browse` añade autorización para examinar los mensajes de la cola (para emitir **MQGET** con la opción browse).
  - `-get` suprime la autorización para obtener (**MQGET**) mensajes de la cola.
  - `+put` añade autorización para transferir (**MQPUT**) mensajes a la cola.

El mandato siguiente revoca la autorización put en la cola MyQueue del principal fvuser y de los grupos groupa y groupb. En sistemas UNIX and Linux, este mandato también revoca la autorización de transferencia (put) para todos los principales del mismo grupo primario que fvuser.

```
setmqaut -m saturn.queue.manager -t queue -n MyQueue -p fvuser
         -g groupa -g groupb -put
```

Utilización del mandato `setmqaut` con un servicio de autorización diferente

Si utiliza su propio servicio de autorización en lugar del OAM, puede especificar el nombre de este servicio en el mandato **setmqaut** para dirigir el mandato a este servicio. Debe especificar este parámetro si tiene varios componentes instalables que se están ejecutando al mismo tiempo; si no es así, la actualización se realiza en el primer componente instalable del servicio de autorización. De forma predeterminada, es el OAM suministrado.

Notas de uso para SET AUTHREC

La lista de autorizaciones a añadir y la lista de autorizaciones a eliminar no se pueden solapar. Por ejemplo, no puede añadir la autorización de visualización y eliminar la autorización de visualización con el mismo mandato. Esta regla se aplica incluso si las autorizaciones se expresan utilizando opciones distintas. Por ejemplo, el mandato siguiente falla porque la autorización DSP se solapa con la autorización ALLADM:

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALLADM)
```

La excepción a este comportamiento de solapamiento es con la autorización ALL. El mandato siguiente añade primero las autorizaciones ALL y, a continuación, elimina la autorización SETID:

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(ALL) AUTHRMV(SETID)
```

El mandato siguiente elimina primero las autorizaciones ALL y, a continuación, añade la autorización DSP:

```
SET AUTHREC PROFILE(*) OBJTYPE(QUEUE) PRINCIPAL(PRINC01) AUTHADD(DSP) AUTHRMV(ALL)
```

Independientemente del orden en el que se proporcionen en el mandato, las ALL se procesan en primer lugar.

ULW Utilización de perfiles genéricos del OAM en UNIX, Linux, and Windows

Utilice perfiles genéricos de OAM para establecer, en una sola operación, los privilegios de un usuario para muchos objetos; en lugar de tener que emitir mandatos **setmqaut** o mandatos **SET AUTHREC** separados para cada objeto individual cuando se crea. Tenga en cuenta que en IBM MQ Appliance solo puede utilizar el mandato **SET AUTHREC** .

Si utiliza perfiles genéricos en los mandatos setmqaut oSET AUTHREC podrá establecer una autorización genérica para todos los objetos que se ajusten a este perfil.

Este conjunto de temas describen de forma más detallada el uso de perfiles genéricos.

Utilización de caracteres comodín en perfiles del OAM

Lo que hace que un perfil sea genérico es el uso de caracteres especiales (caracteres comodín) en el nombre del perfil. Por ejemplo, el carácter comodín de signo de interrogación (?) coincide con cualquier carácter individual de un nombre. Por lo tanto, si especifica ABC.?EF, la autorización que concede a este perfil se aplica a cualquier objeto que tenga los nombres ABC.DEF, ABC.CEF, ABC.BEF, etc.

Los caracteres comodín disponibles son:

?

Utilice el signo de interrogación (?) en lugar de cualquier otro carácter. Por ejemplo, AB.?D se aplica a los objetos AB.CD, AB.ED y AB.FD.

*****

Utilice el asterisco (*) como:

- Un *calificador* de un nombre de perfil para que coincida con cualquier calificador de un nombre de objeto. Un calificador es la parte de un nombre de objeto delimitada por un punto. Por ejemplo, en ABC.DEF.GHI, los calificadores son ABC, DEF y GHI.

  Por ejemplo, ABC.*.JKL se aplica a los objetos ABC.DEF.JKL y ABC.GHI.JKL. (Tenga en cuenta que **no** se aplica a ABC.JKL; cuando el asterisco (*) se utiliza en este contexto siempre indica un calificador.)
- Un carácter contenido en un calificador de un nombre de perfil para que coincida con cero o más caracteres incluidos en el calificador de un nombre de objeto.

  Por ejemplo, ABC.DE*.JKL se aplica a los objetos ABC.DE.JKL, ABC.DEF.JKL y ABC.DEGH.JKL.

******

Utilice el asterisco doble (**) **una vez** en el nombre de un perfil como:

- El nombre de perfil completo para que coincida con todos los nombres de objetos. Por ejemplo, si utiliza `-t prcs` para identificar procesos y, a continuación, utiliza ** como nombre de perfil, puede cambiar las autorizaciones para todos los procesos.

- Como cualquier calificador del principio, mitad o final de un nombre de perfil para que coincida con cero o más calificadores contenidos en un nombre de objeto. Por ejemplo, `**.`ABC identifica todos los objetos con el calificador final ABC.

Sólo puede utilizar el asterisco doble ** como calificador completo:

```
**.DEF
  ABC.**
  A*.**
```

pero no como

```
A**
```

de lo contrario, recibirá el mensaje AMQ7226E: `El nombre de perfil no es válido.`

Nota: Cuando utilice caracteres comodín en los sistemas UNIX y Linux, **debe** colocar el nombre del perfil entre comillas simples.

Prioridades de perfiles

Una cuestión importante que debe comprender cuando utilice perfiles genéricos es la prioridad que se otorga a los perfiles a la hora de decidir qué autorizaciones se han de aplicar a un objeto que se está creando. Por ejemplo, suponga que ha emitido los mandatos:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

El primero otorga autorización de colocación a todas las colas para el principal fred con nombres que coinciden con el perfil AB. *; el segundo otorga autorización de obtención sobre los mismos tipos de cola que coinciden con el perfil AB.C*.

Suponga que ahora crea una cola con el nombre AB.CD. Según las reglas de coincidencia de los caracteres comodín, cualquiera de los mandatos setmqaut se puede aplicar a esta cola. Por lo tanto, la cuestión es ¿tiene autorización para transferir o para obtener?

Para encontrar la respuesta, aplique la regla según la cual cuando varios perfiles pueden aplicarse a un objeto, **sólo se aplica el más específico**. El modo en que se aplica esta regla es comparando los nombres de perfil de izquierda a derecha. Siempre que difieren, un carácter no genérico es más especifico que un carácter genérico. De este modo, en este ejemplo, la cola AB.CD tiene autorización para **obtener** (AB.C* es más específico que AB.*).

Cuando se comparan caracteres genéricos, el orden de *especificidad* es el siguiente:

1. ?
2. *
3. **

Volcado de valores de perfil

Para obtener una definición completa del mandato de control **dmpmqaut** y su sintaxis, consulte dmpmqaut.

Para obtener una definición completa del mandato MQSC **DISPLAY AUTHREC** y su sintaxis, consulte DISPLAY AUTHREC.

Para obtener una definición completa del mandato de PCF **MQCMD_INQUIRE_AUTH_RECS** y su sintaxis, consulte Consultar registros de autorización.

En los ejemplos siguientes se muestra el uso del mandato de control **dmpmqaut** para volcar registros de autorización para perfiles genéricos:

1. En este ejemplo se vuelcan todos los registros de autorización que tienen un perfil que coincide con la cola a.b.c del principal user1.

```
dmpmqaut -m qm1 -n a.b.c -t q -p user1
```

El volcado resultante es similar al siguiente:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

Nota: Aunque los usuarios de UNIX y Linux pueden utilizar la opción `-p` para el mandato **dmpmqaut**, deben utilizar `-g groupname` al definir las autorizaciones.

2. Este ejemplo realiza un volcado de todos los registros de autorización que tienen un perfil que coincide con la cola a.b.c.

```
dmpmqaut -m qmgr1 -n a.b.c -t q
```

El volcado resultante es similar al siguiente:

```
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 ejemplo vuelca todos los registros de autorización para el perfil a.b. *, de tipo cola.

```
dmpmqaut -m qmgr1 -n a.b.* -t q
```

El volcado resultante es similar al siguiente:

```
profile:     a.b.*
object type: queue
entity:      user1
type:        principal
authority:   get, browse, put, inq
```

4. Este ejemplo realiza un volcado de todos los registros de autorización para el gestor de colas qmX.

```
dmpmqaut -m qmX
```

El volcado resultante es similar al siguiente:

```
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 ejemplo realiza un volcado de todos los nombres de perfil y tipos de objeto para el gestor de colas qmX.

   ```
   dmpmqaut -m qmX -l
   ```

   El volcado resultante es similar al siguiente:

   ```
   profile: q1, type: queue
   profile: q*, type: queue
   profile: name.*, type: namelist
   profile: pr1, type: process
   ```

Nota: Solamente para IBM MQ for Windows, todos los principales visualizados incluyen información de dominio, por ejemplo:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

ULW *Utilización de caracteres comodín en perfiles del OAM en UNIX, Linux, and Windows*

Utilice caracteres comodín en un nombre de perfil del gestor de autorizaciones sobre objetos (OAM) para hacer que dicho perfil sea aplicable a más de un objeto.

Lo que hace que un perfil sea genérico es el uso de caracteres especiales (caracteres comodín) en el nombre del perfil. Por ejemplo, el carácter comodín de signo de interrogación (?) coincide con cualquier carácter individual de un nombre. Por lo tanto, si especifica `ABC.?EF`, la autorización que concede a este perfil se aplica a cualquier objeto que tenga los nombres `ABC.DEF`, `ABC.CEF`, `ABC.BEF`, etc.

Los caracteres comodín disponibles son:

?

Utilice el signo de interrogación (?) en lugar de cualquier otro carácter. Por ejemplo, `AB.?D` se aplica a los objetos `AB.CD`, `AB.ED` y `AB.FD`.

*****

Utilice el asterisco (*) como:

- Un *calificador* de un nombre de perfil para que coincida con cualquier calificador de un nombre de objeto. Un calificador es la parte de un nombre de objeto delimitada por un punto. Por ejemplo, en `ABC.DEF.GHI`, los calificadores son `ABC`, `DEF` y `GHI`.

  Por ejemplo, `ABC.*.JKL` se aplica a los objetos `ABC.DEF.JKL` y `ABC.GHI.JKL`. (Tenga en cuenta que **no** se aplica a `ABC.JKL`; cuando el asterisco (*) se utiliza en este contexto siempre indica un calificador.)
- Un carácter contenido en un calificador de un nombre de perfil para que coincida con cero o más caracteres incluidos en el calificador de un nombre de objeto.

Por ejemplo, ABC.DE*.JKL se aplica a los objetos ABC.DE.JKL, ABC.DEF.JKL y ABC.DEGH.JKL.

**

Utilice el asterisco doble (**) **una vez** en el nombre de un perfil como:

- El nombre de perfil completo para que coincida con todos los nombres de objetos. Por ejemplo, si utiliza `-t prcs` para identificar procesos y, a continuación, utiliza ** como nombre de perfil, puede cambiar las autorizaciones para todos los procesos.
- Como cualquier calificador del principio, mitad o final de un nombre de perfil para que coincida con cero o más calificadores contenidos en un nombre de objeto. Por ejemplo, **.ABC identifica todos los objetos con el calificador final ABC.

Nota: Cuando utilice caracteres comodín en sistemas UNIX and Linux, **debe** encerrar el nombre de perfil entre comillas simples.

ULW *Prioridades de perfiles en UNIX, Linux, and Windows*

Se puede aplicar más de un perfil genérico a un único objeto. Cuando este sea el caso, se aplica la regla más específica.

Una cuestión importante que debe comprender cuando utilice perfiles genéricos es la prioridad que se otorga a los perfiles a la hora de decidir qué autorizaciones se han de aplicar a un objeto que se está creando. Por ejemplo, suponga que ha emitido los mandatos:

```
setmqaut -n AB.* -t q +put -p fred
setmqaut -n AB.C* -t q +get -p fred
```

El primero otorga autorización de colocación a todas las colas para el principal fred con nombres que coinciden con el perfil AB. *; el segundo otorga autorización de obtención sobre los mismos tipos de cola que coinciden con el perfil AB.C*.

Suponga que ahora crea una cola con el nombre AB.CD. Según las reglas de coincidencia de los caracteres comodín, cualquiera de los mandatos setmqaut se puede aplicar a esta cola. Por lo tanto, la cuestión es ¿tiene autorización para transferir o para obtener?

Para encontrar la respuesta, aplique la regla según la cual cuando varios perfiles pueden aplicarse a un objeto, **sólo se aplica el más específico**. El modo en que se aplica esta regla es comparando los nombres de perfil de izquierda a derecha. Siempre que difieren, un carácter no genérico es más especifico que un carácter genérico. De este modo, en este ejemplo, la cola AB.CD tiene autorización para **obtener** (AB.C* es más específico que AB.*).

Cuando se comparan caracteres genéricos, el orden de *especificidad* es el siguiente:

1. ?
2. *
3. **

Consulte SET AUTHREC para obtener la información equivalente cuando se utiliza este mandatos MQSC.

ULW *Volcado de valores de perfil en UNIX, Linux, and Windows*

Utilice el mandato de control **dmpmqaut** , el mandato MQSC **DISPLAY AUTHREC** o el mandato PCF **MQCMD_INQUIRE_AUTH_RECS** para volcar las autorizaciones actuales asociadas con un perfil especificado. Tenga en cuenta que en IBM MQ Appliance solo puede utilizar el mandato **DISPLAY AUTHREC** .

Para obtener una definición completa del mandato de control **dmpmqaut** y su sintaxis, consulte dmpmqaut.

Para obtener una definición completa del mandato MQSC **DISPLAY AUTHREC** y su sintaxis, consulte DISPLAY AUTHREC.

Para obtener una definición completa del mandato de PCF **MQCMD_INQUIRE_AUTH_RECS** y su sintaxis, consulte Consultar registros de autorización.

En los ejemplos siguientes se muestra el uso del mandato de control **dmpmqaut** para volcar registros de autorización para perfiles genéricos:

1. En este ejemplo se vuelcan todos los registros de autorización que tienen un perfil que coincide con la cola a.b.c del principal user1.

   ```
   dmpmqaut -m qm1 -n a.b.c -t q -p user1
   ```

   El volcado resultante es similar al ejemplo siguiente:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```

   Nota: Los usuarios de UNIX and Linux no pueden utilizar la opción `-p` ; en su lugar, deben utilizar `-g groupname` .
2. Este ejemplo realiza un volcado de todos los registros de autorización que tienen un perfil que coincide con la cola a.b.c.

   ```
   dmpmqaut -m qmgr1 -n a.b.c -t q
   ```

   El volcado resultante es similar al ejemplo siguiente:

   ```
   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 ejemplo vuelca todos los registros de autorización para el perfil a.b. *, de tipo cola.

   ```
   dmpmqaut -m qmgr1 -n a.b.* -t q
   ```

   El volcado resultante es similar al ejemplo siguiente:

   ```
   profile:     a.b.*
   object type: queue
   entity:      user1
   type:        principal
   authority:   get, browse, put, inq
   ```
4. Este ejemplo realiza un volcado de todos los registros de autorización para el gestor de colas qmX.

   ```
   dmpmqaut -m qmX
   ```

   El volcado resultante es similar al ejemplo siguiente:

   ```
   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 ejemplo realiza un volcado de todos los nombres de perfil y tipos de objeto para el gestor de colas qmX.

   ```
   dmpmqaut -m qmX -l
   ```

   El volcado resultante es similar al ejemplo siguiente:

   ```
   profile: q1, type: queue
   profile: q*, type: queue
   profile: name.*, type: namelist
   profile: pr1, type: process
   ```

Nota: Solamente para IBM MQ for Windows, todos los principales visualizados incluyen información de dominio, por ejemplo:

```
profile:     a.b.*
object type: queue
entity:      user1@domain1
type:        principal
authority:   get, browse, put, inq
```

ULW Visualización de los valores de acceso en UNIX, Linux, and Windows

Utilice el mandato de control **dspmqaut** , el mandato MQSC **DISPLAY AUTHREC** o el mandato PCF **MQCMD_INQUIRE_ENTITY_AUTH** para ver las autorizaciones que tiene un principal o grupo específico para un objeto determinado. Tenga en cuenta que, en IBM MQ Appliance, solamente puede utilizar el mandato **DISPLAY AUTHREC**.

El gestor de colas debe estar en ejecución para poder utilizar este mandato. Cuando modifique el acceso de un principal, el OAM reflejará inmediatamente los cambios. Las autorizaciones sólo pueden visualizarse para un grupo o principal cada vez.

Para obtener una definición completa del mandato de control **dmpmqaut** y su sintaxis, consulte dmpmqaut.

Para obtener una definición completa del mandato MQSC **DISPLAY AUTHREC** y su sintaxis, consulte DISPLAY AUTHREC.

Para obtener una definición completa del mandato de PCF **MQCMD_INQUIRE_AUTH_RECS** y su sintaxis, consulte Consultar registros de autorización.

El ejemplo siguiente muestra el uso del mandato de control **dspmqaut** para visualizar las autorizaciones que tiene el grupo GpAdmin para una definición de proceso llamada Annuities que está en gestor de colas QueueMan1.

```
dspmqaut -m QueueMan1 -t process -n Annuities -g GpAdmin
```

ULW Modificación y revocación del acceso a un objeto de IBM MQ en UNIX, Linux, and Windows

Para modificar el nivel de acceso que un usuario o grupo tiene para un objeto, utilice el mandato de control **setmqaut**, el mandato MQSC **DELETE AUTHREC** del mandato PCF**MQCMD_DELETE_AUTH_REC**.

MQ Appliance Tenga en cuenta que en IBM MQ Appliance, solo puede utilizar el mandato **DELETE AUTHREC**.

El proceso de eliminación de un usuario de un grupo se describe en:

- Windows "Creación y gestión de grupos en Windows" en la página 146
- AIX "Creación y gestión de grupos en AIX" en la página 144
- Solaris "Creación y gestión de grupos en Solaris" en la página 145
- Linux "Creación y gestión de grupos en Linux" en la página 144

.

Al ID de usuario que crea un objeto IBM MQ se le otorga autorizaciones de control totales para dicho objeto. Si elimina este ID de usuario del grupo mqm local (o del grupo Administradores en sistemas Windows), no se revocarán estas autorizaciones. Utilice el mandato de control **setmqaut** o el mandato PCF **MQCMD_DELETE_AUTH_REC** para revocar el acceso a un objeto para el ID de usuario que lo ha creado, después de eliminarlo del grupo mqm o del grupo Administradores.

Para obtener una definición completa del mandato de control setmqaut y su sintaxis, consulte setmqaut.

Para obtener una definición completa del mandato MQSC **DELETE AUTHREC** y su sintaxis, consulte DELETE AUTHREC.

Para obtener una definición completa del mandato PCF **MQCMD_DELETE_AUTH_REC** y su sintaxis, consulte Suprimir registro de autorización.

Windows En Windows, a partir de IBM MQ 8.0, puede suprimir las entradas OAM correspondientes a una cuenta de usuario de Windows concreta en cualquier momento utilizando el parámetro **-u** *SID* de **setmqaut**.

Antes de IBM MQ 8.0, tenía que suprimir las entradas OAM correspondientes a una cuenta de usuario de Windows concreta antes de suprimir el perfil de usuario. No se podían eliminar las entradas OAM después de eliminar la cuenta de usuario.

ULW Impedir comprobaciones de acceso de seguridad en los sistemas UNIX, Linux, and Windows

Para desactivar toda la comprobación de seguridad puede inhabilitar el gestor de autorizaciones sobre objetos (OAM). Esta acción podría resultar adecuada en un entorno de prueba. Al haber inhabilitado o eliminado el OAM, no puede añadir un OAM a un gestor de colas existente.

Si decide que no desea realizar comprobaciones de seguridad (por ejemplo, en un entorno de prueba), puede inhabilitar el OAM de dos modos:

- Antes de crear un gestor de colas, establezca la variable de entorno del sistema operativo MQSNOAUT.

Consulte Descripciones de variables de entorno para obtener información sobre las implicaciones de establecer la variable MQSNOAUT y cómo establecer MQSNOAUT en Windows y UNIX.

- Edite el archivo de configuración del gestor de colas para eliminar el servicio.

Si utiliza el mandato **setmqaut** o **dspmqaut** mientras el OAM está inhabilitado, tenga en cuenta los puntos siguientes:

- El OAM no validará el principal, o grupo, especificado lo que significa que el mandato puede aceptar valores no válidos.
- El OAM no realiza comprobaciones de seguridad e indica que todos los principales y grupos están autorizado a realizar todas las operaciones aplicables de objetos.

Aviso: Cuando se elimina un gestor de autorizaciones sobre objetos, no se puede volver a colocar en un gestor de colas existente. Esto se debe a que el OAM debe estar en su sitio cuando se crea el objeto. Para utilizar el OAM de IBM MQ después de haberlo eliminado, reconstruir el gestor de colas.

Conceptos relacionados

Servicios y componentes instalables para UNIX, Linux y Windows

Tareas relacionadas

Configuración de servicios instalables

Referencia relacionada

Información de referencia de servicios instalables

Otorgar el acceso necesario a los recursos

Utilice este tema para determinar qué tareas realizar para aplicar la seguridad a sistemas IBM MQ en UNIX, Linux, Windows, IBM i y z/OS.

Acerca de esta tarea

Durante esta tarea se decide qué acciones son necesarias para aplicar el nivel de seguridad apropiado a los elementos de su instalación IBM MQ. Cada tarea a la que se refiere ofrece instrucciones paso a paso para todas las plataformas.

Procedimiento

1. ¿Necesita limitar el acceso a su gestor de colas a determinados usuarios?
   a) No: No realice ninguna acción más.
   b) Sí: Vaya hasta la siguiente pregunta.
2. ¿Estos usuarios necesitan acceso de administrador parcial sobre un subconjunto de recursos del gestor de colas?
   a) No: Vaya hasta la siguiente pregunta.
   b) Sí: Consulte “Cómo otorgar acceso de administrador parcial sobre un subconjunto de recursos del gestor de colas” en la página 369.
3. ¿Estos usuarios necesitan acceso de administrador total sobre un subconjunto de recursos de gestor de colas?
   a) No: Vaya hasta la siguiente pregunta.
   b) Sí: Consulte “Cómo otorgar acceso de administrador total sobre un subconjunto de recursos del gestor de colas” en la página 378.
4. ¿Estos usuarios necesitan acceso de sólo lectura a todos los recursos del gestor de colas?
   a) No: Vaya hasta la siguiente pregunta.
   b) Sí: Consulte “Otorgar acceso de sólo lectura a todos los recursos de un gestor de colas” en la página 385.
5. ¿Estos usuarios necesitan acceso de administrador total sobre todos los recursos del gestor de colas?

a) No: Vaya hasta la siguiente pregunta.

b) Sí: Consulte “Otorgar acceso administrativo completo a todos los recursos de un gestor de colas” en la página 387.

6. ¿Necesita que las aplicaciones de usuario se conecten con su gestor de colas?

a) No: Inhabilite la conectividad, tal como se describe en “Eliminar la conectividad con el gestor de colas” en la página 388

b) Sí: Consulte “Cómo permitir que las aplicaciones de usuario se conecten con su gestor de colas” en la página 389.

Multi z/OS Cómo otorgar acceso de administrador parcial sobre un subconjunto de recursos del gestor de colas

Necesita otorgar a algunos usuarios acceso parcial de administrador a algunos, pero no todos, de los recursos del gestor de colas. Utilice esta tabla para determinar las acciones que necesita llevar a cabo.

Tabla 69. Cómo otorgar acceso de administrador parcial a un subconjunto de recursos del gestor de colas

Los usuarios necesitan administrar objetos de este tipo	**Realice esta acción**
Colas	Otorgue acceso de administrador parcial a las colas necesarias, tal como se describe en “Otorgar acceso administrativo limitado a algunas colas” en la página 369
Temas	Otorgue acceso de administrador parcial a los temas necesarios, tal como se describe en “Otorgar acceso administrativo limitado a algunos temas” en la página 371
Canales	Otorgue acceso de administrador parcial a los canales necesarios, tal como se describe en “Otorgar acceso administrativo limitado a algunos canales” en la página 372
El gestor de colas	Otorgue acceso de administrador parcial al gestor de colas, tal como se describe en “Otorgar acceso administrativo parcial a un gestor de colas” en la página 373
todos los Procesos	Otorgue acceso de administrador parcial a los procesos necesarios, tal como se describe en “Otorgar acceso administrativo limitado a algunos procesos” en la página 375
Listas de nombres	Otorgue acceso de administrador parcial a las listas de nombres necesarias, tal como se describe en “Otorgar acceso administrativo limitado a algunas listas de nombres” en la página 376
Servicios	Otorgue acceso de administrador parcial a los servicios necesarios, tal como se describe en “Otorgar acceso administrativo limitado a algunos servicios” en la página 377

Otorgar acceso administrativo limitado a algunas colas

Otorgue acceso administrativo parcial a algunas colas en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado a algunas colas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName ReqdAction
  ```

- IBM i
  Para IBM i, emita el mandato siguiente:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS Para z/OS, emita los mandatos siguientes para otorgar acceso a una cola especificada:

  ```
  RDEFINE MQADMIN QMgrName.QUEUE. ObjectProfile UACC(NONE)
  PERMIT QMgrName.QUEUE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Para especificar qué mandatos MQSC puede ejecutar el usuario en la cola, emita los mandatos siguientes para cada mandato MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction. QType UACC(NONE)
  PERMIT QMgrName. ReqdAction. QType CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Para permitir al usuario utilizar el mandato DISPLAY QUEUE, emita los mandatos siguientes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY. QType UACC(NONE)
  PERMIT QMgrName.DISPLAY. QType CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

  ReqdAction
  La acción que va a permitir realizar al grupo:

– ULW En los sistemas UNIX, Linux, and Windows, cualquier combinación de las autorizaciones siguientes: +chg, +clr, +dlt, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

– IBM i En sistemas IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMDLT, *ADMDSP. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.

– z/OS En z/OS, uno de los valores ALTER, CLEAR, DELETE, o MOVE.

Nota: Otorgar +crt para las colas convierte indirectamente al usuario o grupo en un administrador. No utilice la autorización +crt para otorgar acceso administrativo limitado a algunas colas.

QType

Para el mandato DISPLAY, uno de los valores QUEUE, QLOCAL, QALIAS, QMODEL, QREMOTE o QCLUSTER.

Para otros valores de *AcciónReq*, uno de los valores QLOCAL, QALIAS, QMODEL o QREMOTE.

Otorgar acceso administrativo limitado a algunos temas

Otorgue acceso administrativo parcial a algunos temas en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado a algunos temas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName ReqdAction
```

- IBM i
  Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQADMIN QMgrName.TOPIC. ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Estos mandatos otorgan acceso al tema especificado. Para determinar qué mandatos MQSC puede realizar el usuario en el tema, emita los mandatos siguientes para cada mandato MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.TOPIC UACC(NONE)
PERMIT QMgrName. ReqdAction.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Para permitir al usuario utilizar el mandato DISPLAY TOPIC, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.TOPIC UACC(NONE)
PERMIT QMgrName.DISPLAY.TOPIC CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

ReqdAction
La acción que va a permitir realizar al grupo:

- ULW En sistemas UNIX, Linux, and Windows , cualquier combinación de las autorizaciones siguientes: + chg, + clr, + crt, + dlt, + dsp. + ctrl. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.
- IBM i En IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.
- z/OS En z/OS, uno de los valores ALTER, CLEAR, DEFINE, DELETE o MOVE.

Otorgar acceso administrativo limitado a algunos canales

Otorgue acceso administrativo parcial a algunos canales en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado a algunos canales, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName ReqdAction
```

- IBM i
  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*CHL) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS En z/OS:

```
RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
PERMIT QMgrName.CHANNEL. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Estos mandatos otorgan acceso al canal especificado. Para determinar qué mandatos MQSC puede realizar el usuario en el canal, emita los mandatos siguientes para cada mandato MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.CHANNEL UACC(NONE)
PERMIT QMgrName. ReqdAction.CHANNEL CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Para permitir al usuario utilizar el mandato DISPLAY CHANNEL, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.CHANNEL UACC(NONE)
PERMIT QMgrName.DISPLAY.CHANNEL CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

ReqdAction
La acción que va a permitir realizar al grupo:

- ULW En UNIX, Linux, and Windows, cualquier combinación de las autorizaciones siguientes: + chg, + clr, + crt, + dlt, + dsp. + ctrl, + ctrlx. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.
- IBM i En IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLX. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.
- z/OS En z/OS, uno de los valores ALTER, CLEAR, DEFINE, DELETE o MOVE.

Otorgar acceso administrativo parcial a un gestor de colas

Otorgue acceso administrativo parcial a un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado para llevar a cabo algunas acciones en el gestor de colas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW

  En UNIX, Linux, and Windows:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName ReqdAction
  ```

- IBM i

  En IBM i:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS

  En z/OS:

  Para determinar qué mandatos MQSC puede realizar en el gestor de colas, emita los mandatos siguientes para cada mandato MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.QMGR UACC(NONE)
  PERMIT QMgrName. ReqdAction.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Para permitir al usuario utilizar el mandato DISPLAY QMGR, emita los mandatos siguientes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.QMGR UACC(NONE)
  PERMIT QMgrName.DISPLAY.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

  ReqdAction
  La acción que va a permitir realizar al grupo:

  – ULW En UNIX, Linux, and Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

    Aunque +set es una autorización de MQI y no normalmente no se considera administrativa, otorgar +set en el gestor de colas puede conducir indirectamente a la autorización administrativa completa. No otorgue +set a usuarios y aplicaciones ordinarios.

– IBM i En sistemas IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.

Otorgar acceso administrativo limitado a algunos procesos

Otorgue acceso administrativo parcial a algunos procesos en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado a algunos procesos, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName ReqdAction
  ```

- IBM i
  En IBM i:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(ReqdAction) MQMNAME('
  QMgrName ')
  ```

- z/OS En z/OS:

  ```
  RDEFINE MQADMIN QMgrName.PROCESS. ObjectProfile UACC(NONE)
  PERMIT QMgrName.PROCESS. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Estos mandatos otorgan acceso al canal especificado. Para determinar qué mandatos MQSC puede realizar el usuario en el canal, emita los mandatos siguientes para cada mandato MQSC:

  ```
  RDEFINE MQCMDS QMgrName. ReqdAction.PROCESS UACC(NONE)
  PERMIT QMgrName. ReqdAction.PROCESS CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
  ```

  Para permitir al usuario utilizar el mandato DISPLAY PROCESS, emita los mandatos siguientes:

  ```
  RDEFINE MQCMDS QMgrName.DISPLAY.PROCESS UACC(NONE)
  PERMIT QMgrName.DISPLAY.PROCESS CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

ReqdAction
La acción que va a permitir realizar al grupo:

- ULW En UNIX, Linux, and Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.
- IBM i En sistemas IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.
- z/OS En z/OS, uno de los valores ALTER, CLEAR, DEFINE, DELETE o MOVE.

Otorgar acceso administrativo limitado a algunas listas de nombres

Otorgue acceso administrativo parcial a algunas listas de nombres en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo limitado a algunas listas de nombres, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName ReqdAction
```

- IBM i
  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*NMLIST) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS En z/OS:

```
RDEFINE MQADMIN QMgrName.NAMELIST. ObjectProfile UACC(NONE)
PERMIT QMgrName.NAMELIST. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Estos mandatos otorgan acceso a la lista de nombres especificada. Para determinar qué mandatos MQSC puede realizar el usuario en la lista de nombres, emita los mandatos siguientes para cada mandato MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.NAMELIST UACC(NONE)
PERMIT QMgrName. ReqdAction.NAMELIST CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

Para permitir al usuario utilizar el mandato DISPLAY NAMELIST, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.NAMELIST UACC(NONE)
PERMIT QMgrName.DISPLAY.NAMELIST CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

ReqdAction
La acción que va a permitir realizar al grupo:

- ULW En UNIX, Linux, and Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +ctrl, +ctrlx, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.
- IBM i En IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLX. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.
- z/OS En z/OS, uno de los valores ALTER, CLEAR, DEFINE, DELETE o MOVE.

Otorgar acceso administrativo limitado a algunos servicios

Otorgue acceso administrativo parcial a algunos servicios en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso de administrador limitado a algunas acciones, utilice los mandatos apropiados para su sistema operativo. z/OS Tenga en cuenta que los objetos de servicio no existen en z/OS.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW

En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName ReqdAction
```

- En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*SVC) USER(GroupName) AUT(ReqdAction) MQMNAME('
QMgrName ')
```

- z/OS En z/OS:

  Estos mandatos otorgan acceso al servicio especificado. Para determinar qué mandatos MQSC puede realizar el usuario en el servicio, emita los mandatos siguientes para cada mandato MQSC:

```
RDEFINE MQCMDS QMgrName. ReqdAction.SERVICE UACC(NONE)
PERMIT QMgrName. ReqdAction.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(ALTER)
```

  Para permitir al usuario utilizar el mandato DISPLAY SERVICE, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.DISPLAY.SERVICE UACC(NONE)
PERMIT QMgrName.DISPLAY.SERVICE CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

  ReqdAction
  La acción que va a permitir realizar al grupo:

  - ULW En los sistemas UNIX, Linux, and Windows, cualquier combinación de las autorizaciones siguientes: +chg, +clr, +crt, +dlt, +ctrl, +ctrlx, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.
  - IBM i En IBM i, cualquier combinación de las siguientes autorizaciones: *ADMCHG, *ADMCLR, *ADMCRT, *ADMDLT, *ADMDSP, *CTRL, *CTRLX. La autorización *ALLADM es equivalente a todas estas autorizaciones individuales.

Cómo otorgar acceso de administrador total sobre un subconjunto de recursos del gestor de colas

Necesita otorgar a algunos usuarios acceso completo de administrador a algunos, pero no todos, de los recursos del gestor de colas. Utilice estas tablas para determinar las acciones que necesita llevar a cabo.

Tabla 70. Cómo otorgar acceso de administrador total a un subconjunto de recursos del gestor de colas

Los usuarios necesitan administrar objetos de este tipo	**Realice esta acción**
Colas	Otorgue acceso de administrador total a las colas necesarias, tal como se describe en "Otorgar acceso administrativo completo a algunas colas" en la página 379

Tabla 70. Cómo otorgar acceso de administrador total a un subconjunto de recursos del gestor de colas (continuación)	
Los usuarios necesitan administrar objetos de este tipo	**Realice esta acción**
Temas	Otorgue acceso de administrador total a los temas necesarios, tal como se describe en "Otorgar acceso administrativo completo a algunos temas" en la página 380
Canales	Otorgue acceso de administrador total a los canales necesarios, tal como se describe en "Otorgar acceso administrativo completo a algunos canales" en la página 381
El gestor de colas	Otorgue acceso de administrador total al gestor de colas, tal como se describe en "Otorgar acceso administrativo completo a un gestor de colas" en la página 382
todos los Procesos	Otorgue acceso de administrador total a los procesos necesarios, tal como se describe en "Otorgar acceso administrativo completo a algunos procesos" en la página 383
Listas de nombres	Otorgue acceso de administrador total a las listas de nombres necesarias, tal como se describe en "Otorgar acceso administrativo completo a algunas listas de nombres" en la página 384
Servicios	Otorgue acceso de administrador total a los servicios necesarios, tal como se describe en "Otorgar acceso administrativo completo a algunos servicios" en la página 384

Otorgar acceso administrativo completo a algunas colas

Otorgue acceso administrativo completo a algunas colas en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunas colas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +alladm
```

- IBM i
  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*ALLADM) MQMNAME(' QMgrName
')
```

- z/OS
  En z/OS:

```
RDEFINE MQADMIN QMgrName.QUEUE. ObjectProfile UACC(NONE)
PERMIT QMgrName.QUEUE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a algunos temas

Otorgue acceso administrativo completo a algunos temas en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunos temas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +alladm
```

- IBM i
  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(ALLADM) MQMNAME('
QMgrName ')
```

- z/OS

  En z/OS:

```
RDEFINE MQADMIN QMgrName.TOPIC. ObjectProfile UACC(NONE)
PERMIT QMgrName.TOPIC. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a algunos canales

Otorgue acceso administrativo completo a algunos canales en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunos canales, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW

  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName +alladm
```

- IBM i

  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*CHL) USER(GroupName) AUT(ALLADM) MQMNAME('
QMgrName ')
```

- z/OS

  En z/OS:

```
RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
PERMIT QMgrName.CHANNEL. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a un gestor de colas

Otorgue acceso administrativo completo a un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo al gestor de colas, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

  ```
  setmqaut -m QMgrName -t qmgr -g GroupName +alladm
  ```

- IBM i
  En IBM i:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS
  En z/OS:

  ```
  RDEFINE MQADMIN QMgrName.QMGR UACC(NONE)
  PERMIT QMgrName.QMGR CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a algunos procesos

Otorgue acceso administrativo completo a algunos procesos en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunos procesos, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +alladm
```

- IBM i
  En IBM i:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(*ALLADM) MQMNAME('
QMgrName ')
```

- z/OS
  En z/OS:

```
RDEFINE MQADMIN QMgrName.CHANNEL. ObjectProfile UACC(NONE)
PERMIT QMgrName.PROCESS. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a algunas listas de nombres

Otorgue acceso administrativo completo a algunas listas de nombres en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunas listas de nombres, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW

  En UNIX, Linux, and Windows:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName +alladm
  ```

- IBM i

  En IBM i:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS

  En z/OS:

  ```
  RDEFINE MQADMIN QMgrName.NAMELIST. ObjectProfile UACC(NONE)
  PERMIT QMgrName.NAMELIST. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a algunos servicios

Otorgue acceso administrativo completo a algunos servicios en un gestor de colas, a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar acceso administrativo completo a algunos servicios, utilice los mandatos apropiados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW

  En UNIX, Linux, and Windows:

  ```
  setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName +alladm
  ```

- IBM i

  En IBM i:

  ```
  GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*SVC) USER(GroupName) AUT(*ALLADM) MQMNAME('
  QMgrName ')
  ```

- z/OS

  En z/OS:

  ```
  RDEFINE MQADMIN QMgrName.SERVICE. ObjectProfile UACC(NONE)
  PERMIT QMgrName.SERVICE. ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
  ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso de sólo lectura a todos los recursos de un gestor de colas

Otorgue acceso de sólo lectura a todos los recursos de un gestor de colas para cada usuario o grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Utilice el asistente para añadir autorizaciones basadas en funciones o los mandatos correspondientes para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i

- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Después de haber cambiado los detalles de autorización, realice una renovación de seguridad utilizando el mandato REFRESH SECURITY .

Procedimiento

- Utilización del asistente:
  a) En el panel del navegador de IBM MQ Explorer, pulse con el botón derecho del ratón en el gestor de colas y pulse **Autorizaciones de objetos** > **Añadir autorizaciones basadas en funciones**
  Se abre el asistente Añadir autorizaciones basadas en funciones.
- Windows UNIX
  Para sistemas UNIX y Windows, emita los siguientes mandatos:

```
setmqaut -m QMgrName -n ** -t queue -g GroupName +browse +dsp
setmqaut -m QMgrName -n SYSTEM.ADMIN.COMMAND.QUEUE -t queue -g GroupName +dsp +inq +put
setmqaut -m QMgrName -n SYSTEM.MQEXPLORER.REPLY.MODEL -t queue -g GroupName +dsp +inq +get
+put
setmqaut -m QMgrName -n ** -t topic -g GroupName +dsp
setmqaut -m QMgrName -n ** -t channel -g GroupName +dsp +inq
setmqaut -m QMgrName -n ** -t clntconn -g GroupName +dsp
setmqaut -m QMgrName -n ** -t authinfo -g GroupName +dsp
setmqaut -m QMgrName -n ** -t listener -g GroupName +dsp
setmqaut -m QMgrName -n ** -t namelist -g GroupName +dsp
setmqaut -m QMgrName -n ** -t process -g GroupName +dsp
setmqaut -m QMgrName -n ** -t service -g GroupName +dsp
setmqaut -m QMgrName -t qmgr -g GroupName +dsp +inq +connect
```

  Las autorizaciones específicas para SYSTEM.ADMIN.COMMAND.QUEUE y SYSTEM.MQEXPLORER.REPLY.MODEL sólo es necesario si desea utilizar IBM MQ Explorer.
- IBM i
  Para IBM i, emita los mandatos siguientes:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*Q) USER('GroupName') AUT(*ADMDSP *BROWSE) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*TOPIC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CHL) USER('GroupName') AUT(*ADMDSP *INQ) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*CLTCN) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*AUTHINFO) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*LSR) USER('GroupName') AUT(*ADMDSP)MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*NMLIST) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*PRC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ(*ALL) OBJTYPE(*SVC) USER('GroupName') AUT(*ADMDSP) MQMNAME('QMgrName')
GRTMQMAUT OBJ('object-name') OBJTYPE(*MQM) USER('GroupName') AUT(*ADMDSP *CONNECT *INQ)
MQMNAME('QMgrName')
```

- z/OS
  Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQQUEUE QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQQUEUE) ID(GroupName) ACCESS(READ)
RDEFINE MXTOPIC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MXTOPIC) ID(GroupName) ACCESS(READ)
RDEFINE MQPROC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQPROC) ID(GroupName) ACCESS(READ)
RDEFINE MQNLIST QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

```
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar acceso administrativo completo a todos los recursos de un gestor de colas

Otorgue acceso administrativo completo a todos los recursos de un gestor de colas para cada usuario o grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Puede utilizar el asistente Añadir autorizaciones basadas en roles o los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Notas: ULW

1. Si utiliza **runmqsc** para administrar el gestor de colas en lugar de IBM MQ Explorer, debe otorgar autorización para consultar, obtener y examinar SYSTEM.MQSC.REPLY.QUEUE, y no es necesario que otorgue ninguna autorización sobre SYSTEM.MQEXPLORER.REPLY.MODEL .
2. Al otorgar a un usuario acceso a todos los recursos de un gestor de colas, hay algunos mandatos que el usuario no puede ejecutar, a menos que dicho usuario tenga acceso de lectura al archivo `qm.ini` . Esto se debe a las restricciones sobre los usuarios que no son de `mqm` que pueden leer el archivo `qm.ini` .

   El usuario no puede emitir los mandatos siguientes a menos que haya otorgado a dicho usuario acceso de lectura al archivo qm.ini :

   - Definición de un canal que está configurado para utilizar TLS
   - Definición de un canal utilizando variables de inserción de configuración automática definidas en `qm.ini`

Procedimiento

- Si está utilizando el asistente, en el panel IBM MQ Explorer Navigator , pulse con el botón derecho del ratón en el gestor de colas y pulse **Autorizaciones de objeto** > **Añadir autorizaciones basadas en roles**.

  Se abre el asistente Añadir autorizaciones basadas en funciones.

- Linux UNIX

  Para sistemas UNIX and Linux, emita los mandatos siguientes:

```
setmqaut -m QMgrName -n '**' -t queue -g GroupName +alladm +browse
setmqaut -m QMgrName -n @class -t queue -g GroupName +crt
setmqaut -m QMgrName -n SYSTEM.ADMIN.COMMAND.QUEUE -t queue -g GroupName +dsp +inq +put
setmqaut -m QMgrName -n SYSTEM.MQEXPLORER.REPLY.MODEL -t queue -g GroupName +dsp +inq +get +put
setmqaut -m QMgrName -n '**' -t topic -g GroupName +alladm
setmqaut -m QMgrName -n @class -t topic -g GroupName +crt
setmqaut -m QMgrName -n '**' -t channel -g GroupName +alladm
setmqaut -m QMgrName -n @class -t channel -g GroupName +crt
setmqaut -m QMgrName -n '**' -t clntconn -g GroupName +alladm
setmqaut -m QMgrName -n @class -t clntconn -g GroupName +crt
setmqaut -m QMgrName -n '**' -t authinfo -g GroupName +alladm
setmqaut -m QMgrName -n @class -t authinfo -g GroupName +crt
setmqaut -m QMgrName -n '**' -t listener -g GroupName +alladm
setmqaut -m QMgrName -n @class -t listener -g GroupName +crt
setmqaut -m QMgrName -n '**' -t namelist -g GroupName +alladm
setmqaut -m QMgrName -n @class -t namelist -g GroupName +crt
setmqaut -m QMgrName -n '**' -t process -g GroupName +alladm
setmqaut -m QMgrName -n @class -t process -g GroupName +crt
setmqaut -m QMgrName -n '**' -t service -g GroupName +alladm
setmqaut -m QMgrName -n @class -t service -g GroupName +crt
setmqaut -m QMgrName -t qmgr -g GroupName +alladm +connect
```

  Consulte **setmqaut** para obtener más información sobre `@class`

- Windows

  Para sistemas Windows, emita los mismos mandatos que para los sistemas UNIX and Linux pero utilizando el nombre de perfil @CLASS en lugar de @class.

- IBM i

  Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*ALL) USER(' GroupName ') AUT(*ALLADM) MQMNAME(' QMgrName ')
```

- z/OS

  Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQADMIN QMgrName.*.** UACC(NONE)
PERMIT QMgrName.*.** CLASS(MQADMIN) ID(GroupName) ACCESS(ALTER)
```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Eliminar la conectividad con el gestor de colas

Si no desea que las aplicaciones de usuario se conecten con el gestor de colas, elimine su autorización para conectarse a él.

Acerca de esta tarea

Revoque la autorización de todos los usuarios a conectarse con el gestor de colas mediante el mandato adecuado para su sistema operativo.

En sistemas UNIX, Linux, Windows, y IBM i, también puede utilizar el mandato DELETE AUTHREC.

Nota: En IBM MQ Appliance solamente puede utilizar el mandato **DELETE AUTHREC**.

Procedimiento

- ULW
  Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

  ```
  setmqaut -m QMgrName -t qmgr -g GroupName -connect
  ```

- IBM i
  Para IBM i, emita el mandato siguiente:

  ```
  RVKMQMAUT OBJ ('QMgrName') OBJTYPE(*MQM) USER(*ALL) AUT(*CONNECT)
  ```

- z/OS
  Para z/OS, emita los mandatos siguientes:

  ```
  RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
  RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
  RDEFINE MQCONN QMgrName.CICS UACC(NONE)
  RDEFINE MQCONN QMgrName.IMS UACC(NONE)
  ```

  No emita ningún mandato PERMIT.

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas.

  z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  GroupName
  Nombre del grupo al que se va a negar el acceso.

Cómo permitir que las aplicaciones de usuario se conecten con su gestor de colas

Desea permitir que la aplicación de usuario se conecte con su gestor de colas. Utilice las tablas de este tema para determinar qué acciones deben llevarse a cabo.

En primer lugar determine si las aplicaciones de cliente se conectarán con su gestor de colas.

Si ninguna de las aplicaciones que se conectarán a su gestor de colas es una aplicación de cliente, inhabilite el acceso remoto tal como se describe en “Inhabilitar el acceso remoto al gestor de colas” en la página 397.

Si una o más de las aplicaciones que se conectarán a su gestor de colas son aplicaciones de cliente, asegure la conectividad remota tal como se describe en “Cómo proteger la conectividad remota con el gestor de colas” en la página 390.

En ambos casos, establezca la seguridad de la conexión tal como se describe en “Configurar la seguridad de conexión” en la página 397

Si desea controlar el acceso a los recursos para cada usuario que se conecta con el gestor de colas, consulte la tabla siguiente. Si la declaración de la primera columna es true, lleve a cabo la acción que aparece en la segunda columna.

Sentencia	Realice esta acción
Tiene aplicaciones que utilizan colas	Consulte “Control del acceso de los usuarios a las colas” en la página 398.
Tiene aplicaciones que utilizan temas	Consulte “Control del acceso de los usuarios a los temas” en la página 405.

Sentencia	Realice esta acción
Tiene aplicaciones que consultan en el objeto del gestor de colas	Consulte “Otorgar autorización para consultar en un gestor de colas” en la página 406.
Tiene aplicaciones que utilizan objetos de procesos	Consulte “Otorgar autorización para acceder a procesos” en la página 407.
Tiene aplicaciones que utilizan listas de nombres	Consulte “Otorgar autorización para acceder a listas de nombres” en la página 408.

Cómo proteger la conectividad remota con el gestor de colas

Puede proteger la conectividad remota con el gestor de colas utilizando TLS, una salida de seguridad, registros de autenticación de canal o una combinación de estos métodos.

Acerca de esta tarea

Puede conectar un cliente con el gestor de colas utilizando un canal de conexión de cliente en la estación de trabajo cliente y un canal de conexión de servidor en el servidor. Proteja estas conexiones de una de las siguientes maneras.

Procedimiento

1. Utilizando TLS con registros de autenticación de canal:
   a) Impida que cualquier Nombre distinguido (DN) abra un canal, utilizando un registro de autenticación de canal SSLPEERMAP para correlacionar todos los DN con USERSRC(NOACCESS).
   b) Permita que Nombres distinguidos (DN) o conjuntos de DNs específicos abran un canal, utilizando un registro de autenticación de canal SSLPEERMAP para correlacionarlos con USERSRC(CANAL).
2. Utilizando TLS con una salida de seguridad:
   a) Establezca MCAUSER en el canal de conexión de servidor en un identificador de usuario sin privilegios.
   b) Escriba una salida de seguridad para asignar un valor MCAUSER en función del valor del DN TLS que reciba en los campos SSLPeerNamePtr y SSLPeerNameLength que se pasan a la salida en la estructura MQCD.
3. Utilizando TLS con valores de definición de canal fijos:
   a) Establezca SSLPEER en el canal de conexión de servidor en un valor o un rango reducido de valores específico.
   b) Establezca MCAUSER en el canal de conexión de servidor en el ID de usuario con el que debe ejecutarse el canal.
4. Utilizando registros de autenticación de canal en canales que no utilizan TLS:
   a) Impida que cualquier dirección IP abra canales, utilizando un registro de autenticación de canal de correlación de direcciones con ADDRESS(*) y USERSRC(NOACCESS).
   b) Permita que direcciones IP específicas abran canales, utilizando registros de autenticación de canal de correlación de direcciones para esas direcciones con USERSRC(CHANNEL).
5. Utilizando una salida de seguridad:
   a) Escriba una salida de seguridad para autorizar conexiones basadas en la propiedad que elija, por ejemplo la dirección IP de origen.
6. También es posible utilizar registros de autenticación de canal con una salida de seguridad, o utilizar los tres métodos, si sus circunstancias específicas lo exigen.

Bloquear direcciones IP específicas

Puede impedir que un canal específico acepte una conexión entrante de una dirección IP o impedir que el gestor de colas en su conjunto permita el acceso desde una dirección IP, utilizando un registro de autenticación de canal.

Antes de empezar

Habilite los registros de autenticación de canal ejecutando el mandato siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Acerca de esta tarea

Para no permitir que canales específicos acepten una conexión de entrada y garantizar que las conexiones sólo se acepten cuando se utilice el nombre de canal correcto, se puede utilizar un tipo de regla para bloquear direcciones IP. Para no permitir que una dirección IP acceda al gestor de colas en su conjunto, lo haría normalmente utilizando un cortafuegos para bloquearla permanentemente. No obstante, se puede utilizar otro tipo de regla para permitirle bloquear unas pocas direcciones temporalmente, por ejemplo mientras espera a que se actualice el cortafuegos.

Procedimiento

- Para impedir que las direcciones IP utilicen un canal específico, establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**.

```
SET CHLAUTH(generic-channel-name) TYPE(ADDRESSMAP) ADDRESS(generic-ip-address)
USERSRC(NOACCESS)
```

  Este mandato tiene tres partes:

  SET CHLAUTH (*nombre-canal-genérico*)
  Esta parte del mandato se utiliza para controlar si desea bloquear una conexión para todo el gestor de colas, un único canal o un rango de canales. Lo que se especifica aquí determina qué áreas se cubren.

  Por ejemplo:

  – SET CHLAUTH('*') - bloquea todos los canales de un gestor de colas, es decir, todo el gestor de colas
  – SET CHLAUTH('SYSTEM.*') - bloque todos los canales que empiezan por SYSTEM.
  – SET CHLAUTH('SYSTEM.DEF.SVRCONN') - bloque el canal SYSTEM.DEF.SVRCONN

  Tipo de regla CHLAUTH
  Utilice esta parte del mandato para especificar el tipo de mandato y determinar si desea proporcionar una sola dirección o una lista de direcciones.

  Por ejemplo:

  – TYPE(ADDRESSMAP) - Use ADDRESSMAP si desea suministrar una dirección única o de comodín. wildcard address. Por ejemplo, ADDRESS('192.168.*') bloquea las conexiones procedentes de una dirección IP que empieza por 192.168.

    Para obtener más información sobre cómo filtrar direcciones IP con patrones, consulte Direcciones IP genéricas.
  – TYPE(BLOCKADDR) - Utilice BLOCKADDR si desea proporcionar una lista de direcciones a bloquear.

  Parámetros adicionales
  Estos parámetros dependen del tipo de regla utilizada en la segunda parte del mandato:

  – Para TYPE(ADDRESSMAP), se utiliza ADDRESS
  – Para TYPE(BLOCKADDR), se utiliza ADDRLIST

Referencia relacionada
SET CHLAUTH

Bloqueo temporal de direcciones IP específicas si el gestor de colas no está en ejecución
Es posible que quiera bloquear direcciones IP específicas, o rangos de direcciones, cuando el gestor de colas no se esté ejecutando y, por lo tanto, no pueda emitir mandatos MQSC. Puede bloquear temporalmente direcciones IP de forma excepcional modificando el archivo `blockaddr.ini`.

Acerca de esta tarea

El archivo `blockaddr.ini` contiene una copia de las definiciones BLOCKADDR que utiliza el gestor de colas. El escucha lee este archivo si se inicia antes que el gestor de colas. En estas circunstancias, el escucha utiliza los valores añadidos manualmente al archivo `blockaddr.ini`.

No obstante, tenga en cuenta que, cuando el gestor de colas se inicia, graba el conjunto de definiciones BLOCKADDR en el archivo `blockaddr.ini`, sobrescribiendo cualquier edición manual que se haya realizado. De forma similar, cada vez que se añade o se suprime una definición BLOCKADDR mediante el mandato **SET CHLAUTH**, el archivo `blockaddr.ini` se actualiza. Por lo tanto, puede realizar cambios permanentes en las definiciones BLOCKADDR sólo mediante el mandato **SET CHLAUTH** cuando el gestor de colas se esté ejecutando.

Procedimiento

1. Abra el archivo `blockaddr.ini` en un editor de texto.
   El archivo se encuentra en el directorio de datos del gestor de colas.
2. Añada direcciones IP como simples pares de palabra clave-valor, donde la palabra clave es `Addr`.
   Si desea información sobre cómo filtrar direcciones IP con patrones, consulte Direcciones IP genéricas.
   Por ejemplo:

```
Addr = 192.0.2.0
Addr = 192.0.*
Addr = 192.0.2.1-8
```

Tareas relacionadas

"Bloquear direcciones IP específicas" en la página 390
Puede impedir que un canal específico acepte una conexión entrante de una dirección IP o impedir que el gestor de colas en su conjunto permita el acceso desde una dirección IP, utilizando un registro de autenticación de canal.

Referencia relacionada

SET CHLAUTH

Bloquear identificadores (ID) de usuario específicos
Puede impedir que usuarios específicos utilicen un canal, especificando identificadores (ID) de usuario que, si se confirman, hacen que el canal finalice. Para ello, establezca un registro de autenticación de canal.

Antes de empezar

Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimiento

Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(BLOCKUSER) USERLIST(userID1, userID2)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
La lista de usuarios proporcionada en un TYPE(BLOCKUSER) sólo se aplica a los canales SVRCONN y no a los canales del gestor de colas al gestor de colas.
IDusuario1 e *IDusuario2* son cada uno el ID de un usuario al que se va a impedir utilizar el canal. También puede también especificar el valor especial *MQADMIN para hacer referencia a los usuarios con privilegios administrativos. Para obtener más información acerca de los usuarios privilegiados, consulte "Usuarios privilegiados" en la página 337. Para obtener más información sobre *MQADMIN, consulte SET CHLAUTH.

Referencia relacionada
SET CHLAUTH

Correlacionar un gestor de colas remoto con un ID de usuario MCAUSER
Puede utilizar un registro de autenticación de canal para establecer el atributo MCAUSER de un canal, según el gestor de colas desde el que se conecta el canal.

Antes de empezar
Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Acerca de esta tarea

Opcionalmente, puede restringir las direcciones IP a las que se aplica la regla.

Tenga en cuenta que esta técnica no se aplica a canales de conexión con el servidor. Si especifica el nombre de un canal de conexión con el servidor en los mandatos siguientes, no tiene ningún efecto.

Procedimiento

- Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

  ```
  SET CHLAUTH(' generic-channel-name ') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
  ) USERSRC(MAP) MCAUSER(user)
  ```

  nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
  nombre-gestcolas-asociado-genérico es el nombre del gestor de colas, o un patrón que incluye el símbolo de asterisco (*) como comodín que coincide con el nombre del gestor de colas.
  usuario es el ID de usuario que se utilizará para todas las conexiones del gestor de colas especificado.
- Para restringir este mandato a determinadas direcciones IP, incluya el parámetro **ADDRESS**, de la siguiente manera:

  ```
  SET CHLAUTH(' generic-channel-name ') TYPE (QMGRMAP) QMNAME(generic-partner-qmgr-name
  ) USERSRC(MAP) MCAUSER(user) ADDRESS(
  generic-ip-address)
  ```

  nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
  dirección-ip-genérica es una dirección individual, o un patrón que incluye el símbolo asterisco (*) como comodín o el guión (-) para indicar un rango, que coincide con la dirección. Si desea más información sobre direcciones IP genéricas, consulte Direcciones IP genéricas.

Referencia relacionada
SET CHLAUTH

Correlación de un ID de usuario cliente con un ID de usuario MCAUSER
Se puede utilizar un registro de autenticación de canal para cambiar el atributo MCAUSER de un canal de conexión con el servidor en función del ID de usuario recibido de un cliente.

Antes de empezar
Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Acerca de esta tarea
Tenga en cuenta que esta técnica sólo se aplica a canales de conexión con el servidor. No tiene ningún efecto en otros tipos de canal.

Procedimiento
Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record** . Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE (USERMAP) CLNTUSER(client-user-name) USERSRC(MAP)
MCAUSER(
user)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
nombre-usuario-cliente es el ID de usuario asociado a la conexión de cliente, el valor podría ser confirmado por la aplicación cliente, modificado por autenticación de conexión usando una adopción temprana o establecido vía salida de canal.
usuario es el ID de usuario que se utilizará en lugar del nombre de usuario del cliente.

Referencia relacionada
SET CHLAUTH
Atributos de la stanza de canales (ChlauthEarlyAdopt)

Correlacionar un Nombre distinguido SSL o TLS con un ID de usuario MCAUSER
Puede utilizar un registro de autenticación de canal para establecer el atributo MCAUSER de un canal, según el Nombre distinguido (DN) recibido.

Antes de empezar
Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimiento
Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE (SSLPEERMAP)
SSLPEER(generic-ssl-peer-name) SSLCERTI(generic-issuer-name)
USERSRC(MAP) MCAUSER(user)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
nombre-igual-ssl-genérico es una serie que sigue las reglas de IBM MQ estándar para los valores de SSLPEER. Consulte Reglas de IBM MQ para valores SSLPEER.
usuario es el ID de usuario que se utilizará para todas las conexiones que utilicen el DN especificado.
nombre-emisor-genérico hace referencia al DN del emisor del certificado que ha de coincidir. Este parámetro opcional pero debe utilizarlo para evitar que el certificado erróneo coincida falsamente si se utilizan varias entidades emisoras de certificados.

Referencia relacionada
SET CHLAUTH

Bloquear el acceso desde un gestor de colas remoto
Puede utilizar un registro de autenticación de canal para impedir que un gestor de colas remoto inicie canales.

Antes de empezar
Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Acerca de esta tarea
Tenga en cuenta que esta técnica no se aplica a canales de conexión con el servidor. Si especifica el nombre de un canal de conexión con el servidor en el mandato siguiente, no tiene ningún efecto.

Procedimiento
Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(QMGRMAP) QMNAME(' generic-partner-qmgr-name ')
USERSRC(NOACCESS)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
nombre-gestcolas-asociado-genérico es el nombre del gestor de colas, o un patrón que incluye el símbolo de asterisco (*) como comodín que coincide con el nombre del gestor de colas.

Referencia relacionada
SET CHLAUTH

Bloqueo del acceso de un ID de usuario cliente
Se puede utilizar un registro de autenticación de canal para impedir que un ID de usuario establezca una conexión de canal.

Antes de empezar
Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Acerca de esta tarea
Tenga en cuenta que esta técnica sólo se aplica a canales de conexión con el servidor. No tiene ningún efecto en otros tipos de canal.

Procedimiento

Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(USERMAP) CLNTUSER(' client-user-name ')
USERSRC(NOACCESS)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
nombre-usuario-cliente es el ID de usuario asociado a la conexión de cliente, el valor podría ser confirmado por la aplicación cliente, modificado por autenticación de conexión usando una adopción temprana o establecido vía salida de canal.

Referencia relacionada
SET CHLAUTH

Bloquear el acceso para un Nombre distinguido SSL o TLS
Puede utilizar un registro de autenticación de canal para impedir que un Nombre distinguido (DN) TLS inicie canales.

Antes de empezar

Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimiento

Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH('generic-channel-name') TYPE(SSLPEERMAP)
SSLPEER('generic-ssl-peer-name') SSLCERTI(generic-issuer-name)
USERSRC(NOACCESS)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
nombre-igual-ssl-genérico es una serie que sigue las reglas de IBM MQ estándar para los valores de SSLPEER. Consulte Reglas de IBM MQ para valores SSLPEER.
nombre-emisor-genérico hace referencia al DN del emisor del certificado que ha de coincidir. Este parámetro opcional pero debe utilizarlo para evitar que el certificado erróneo coincida falsamente si se utilizan varias entidades emisoras de certificados.

Referencia relacionada
SET CHLAUTH

Correlacionar una dirección IP con un ID de usuario MCAUSER
Puede utilizar un registro de autenticación de canal para establecer el atributo MCAUSER de un canal, según la dirección IP desde la que se recibe la conexión.

Antes de empezar

Asegúrese de que los registros de canal de autenticación estén habilitados del modo siguiente:

```
ALTER QMGR CHLAUTH(ENABLED)
```

Procedimiento

Establezca un registro de autenticación de canal utilizando el mandato MQSC **SET CHLAUTH**o el mandato PCF **Set Channel Authentication Record**. Por ejemplo, puede emitir el mandato MQSC:

```
SET CHLAUTH(' generic-channel-name ') TYPE(ADDRESSMAP) ADDRESS(' generic-ip-address ')
USERSRC(MAP) MCAUSER(user)
```

nombre-canal-genérico es el nombre de un canal para el que desea controlar el acceso, o un patrón que incluye el símbolo asterisco (*) como un comodín que coincide con el nombre de canal.
usuario es el ID de usuario que se utilizará para todas las conexiones que utilicen el DN especificado.
dirección-ip-genérica es la dirección desde la que se establece la conexión, o un patrón que incluye el asterisco (*) como comodín o el guión (-) para indicar un rango, que coincide con la dirección.

Referencia relacionada
SET CHLAUTH

Inhabilitar el acceso remoto al gestor de colas

Si no desea que las aplicaciones cliente se conecten con su gestor de colas, inhabilite el acceso remoto a ellas.

Acerca de esta tarea

Evite que las aplicaciones clientes se conecten al gestor de colas de una de las maneras siguientes:

Procedimiento

- Suprima todos los canales de conexión con el servidor utilizando el mandato MQSC **DELETE CHANNEL**.
- Establezca como identificador de usuario del agente del canal de mensajes (MCAUSER) del canal un ID de usuario sin derecho de acceso, mediante el mandato MQSC **ALTER CHANNEL**.

Configurar la seguridad de conexión

Otorgue la autorización para conectarse con el gestor de colas a cada usuario o grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para configurar la seguridad de conexión, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  En UNIX, Linux, and Windows:

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

- IBM i

  En IBM i:

  ```
  GRTMQMAUT OBJ('QMgrName') OBJTYPE(*MQM) USER('GroupName') AUT(*CONNECT)
  ```

- z/OS

  En z/OS:

  ```
  RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
  PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
  RDEFINE MQCONN QMgrName.CICS UACC(NONE)
  PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
  RDEFINE MQCONN QMgrName.IMS UACC(NONE)
  PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
  RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
  PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
  ```

  Estos mandatos otorgan autorización de conexión para el lote, CICS, IMS y el iniciador de canal (CHIN). Si no utiliza un tipo concreto de conexión, omita los mandatos correspondientes.

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Conceptos relacionados

"Perfiles de seguridad de conexión para el iniciador de canal" en la página 200
Los perfiles para comprobar las conexiones del iniciador de canal están formados por el nombre del gestor de colas o del grupo de compartición de colas, seguido de la palabra *CHIN*. Otorgue al ID de usuario utilizado por el espacio de direcciones de tareas iniciadas del iniciador de canal acceso de lectura (READ) al perfil de conexión.

Control del acceso de los usuarios a las colas

Desea controlar el acceso de la aplicación a las colas. Utilice este tema para determinar qué acciones deben llevarse a cabo.

Para cada declaración true de la primera columna, lleve a cabo la acción indicada en la segunda columna.

Sentencia	Acción
La aplicación obtiene menajes de una cola	Consulte "Otorgar autorización para obtener mensajes de colas" en la página 399.
La aplicación establece el contenido	Consulte "Otorgar autorización para establecer contexto" en la página 399.
La aplicación pasa el contexto	Consulte "Otorgar autorización para pasar contexto" en la página 401.
La aplicación transfiere mensajes a una cola agrupada en clúster	Consulte "Autorización de transferencia de mensajes a colas de clústeres remotos" en la página 466.
La aplicación transfiere mensajes a una cola local	Consulte "Otorgar autorización para transferir mensajes a una cola local" en la página 402.

Sentencia	Acción
La aplicación transfiere mensajes a una cola modelo	Consulte “Otorgar autorización para transferir mensajes a una cola modelo” en la página 402.
La aplicación transfiere mensajes a una cola remota	Consulte “Otorgar autorización para transferir mensajes a una cola de clúster remota” en la página 403.

Otorgar autorización para obtener mensajes de colas
Otorgue la autorización para obtener mensajes de una cola o un conjunto de colas a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para obtener mensajes de algunas colas locales, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +get
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*GET) MQMNAME(' QMgrName ')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQADMIN) ID(GroupName) ACCESS(UPDATE)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para establecer contexto
Otorgue la autorización para establecer contexto en un mensaje recibido que se está transfiriendo a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para establecer contexto en algunas colas, utilice los mandatos adecuados de para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita uno de los mandatos siguientes:
  - Para establecer sólo contexto de identidad:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setid
    ```

  - Para establecer todo el contexto:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +setall
    ```

  Nota: Para utilizar las autoridades `setid` o `setall`, hay que otorgar las autoridades al correspondiente objeto de cola y también al objeto de gestor de colas.
- Para IBM i, emita uno de los siguientes mandatos:
  - Para establecer sólo contexto de identidad:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*SETID) MQMNAME('
    QMgrName ')
    ```

  - Para establecer todo el contexto:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*SETALL) MQMNAME('
    QMgrName ')
    ```

- Para z/OS, emita uno de los siguientes conjuntos de mandatos:
  - Para establecer sólo contexto de identidad:

    ```
    RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
    PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
    ```

  - Para establecer todo el contexto:

    ```
    RDEFINE MQQUEUE QMgrName. ObjectProfile UACC(NONE)
    PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(CONTROL)
    ```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para pasar contexto
Otorgue la autorización para pasar contexto de un mensaje recibido a uno que se está transfiriendo a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para pasar contexto a algunas colas, utilice los mandatos adecuados de para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- ULW
  Para sistemas UNIX, Linux, and Windows, emita uno de los mandatos siguientes:
  - Para pasar sólo contexto de identidad:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passid
    ```

  - Para pasar todo el contexto:

    ```
    setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +passall
    ```

- IBM i
  Para IBM i, emita uno de los siguientes mandatos:
  - Para pasar sólo contexto de identidad:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PASSID) MQMNAME('
    QMgrName ')
    ```

  - Para pasar todo el contexto:

    ```
    GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PASSALL) MQMNAME('
    QMgrName ')
    ```

- z/OS
  Para z/OS, emita los siguientes mandatos para pasar contexto de identidad o todo el contexto:

  ```
  RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
  PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
  ```

  Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para transferir mensajes a una cola local
Otorgue la autorización para transferir mensajes a una cola local o a un conjunto de colas a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para transferir mensajes a algunas colas locales, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQQUEUE) ID(GroupName) ACCESS(UPDATE)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para transferir mensajes a una cola modelo
Otorgue la autorización para transferir mensajes a una cola modelo o a un conjunto de colas modelo a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Las colas modelo se utilizan para crear colas dinámicas. Por lo tanto, debe otorgar autorización a la colas modelo y dinámicas. Para otorgar estas autorizaciones, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita los mandatos siguientes:

```
setmqaut -m QMgrName -n ModelQueueName -t queue -g GroupName +put
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +put
```

- Para IBM i, emita los mandatos siguientes:

```
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 los mandatos siguientes:

```
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)
```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  NombreColaModelo
  El nombre de la cola modelo en la que se basan las colas dinámicas.

  ObjectProfile
  El nombre de la cola dinámica o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para transferir mensajes a una cola de clúster remota
Otorgue la autorización para transferir mensajes a una cola de clúster remota o a un conjunto de colas a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para colocar un mensaje en una cola de clúster remota puede ponerlo en una definición local de una cola remota o en un cola remota con nombre completo. Si está utilizando una definición local de una cola remota, necesitará la autoridad para transferir el objeto local: consulte “Otorgar autorización para transferir mensajes a una cola local” en la página 402. Si utiliza un completo de la cola remota,

necesitará autorización para transferir a la cola remota. Otorgue esta autorización mediante los mandatos adecuados para su sistema operativo.

El comportamiento predeterminado es realizar el control de acceso para SYSTEM.CLUSTER.TRANSMIT.QUEUE. Tenga en cuenta que este comportamiento se aplica, incluso si está utilizando varias colas de transmisión.

El comportamiento descrito en este tema solamente se aplica si ha configurado el atributo **ClusterQueueAccessControl** en el archivo qm.ini para que sea *RQMName*, tal como se describe en el tema Stanza de seguridad y si ha reiniciado el gestor de colas.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -t rqmname -n
ObjectProfile -g GroupName +put
```

  Tenga en cuenta que puede utilizar el objeto *rqmname* para las colas de clúster remoto sólo.
- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJTYPE(*RMTMQMNAME) OBJ('
ObjectProfile') USER(GroupName) AUT(*PUT) MQMNAME('
QMgrName')
```

  Tenga en cuenta que puede utilizar el objeto RMTMQMNAME para las colas de clúster remoto sólo.
- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQQUEUE QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.QUEUE.ObjectProfile CLASS(MQQUEUE)
ID(GroupName) ACCESS(UPDATE)
```

  Tenga en cuenta que puede usar el nombre del gestor de colas remoto (o el grupo de compartición de colas) solo para las colas de clúster remoto.

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del gestor de colas remoto o el perfil genérico para el cual se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Control del acceso de los usuarios a los temas

Necesita controlar el acceso de las aplicaciones a los temas. Utilice este tema para determinar qué acciones deben llevarse a cabo.

Para cada declaración true de la primera columna, lleve a cabo la acción indicada en la segunda columna.

Tabla 71. Control del acceso de los usuarios a los temas

Sentencia	**Acción**
La aplicación publica mensajes en un tema	Consulte "Otorgar autorización para publicar mensajes en un tema" en la página 405.
La aplicación se suscribe a un tema	Consulte "Otorgar autorización para suscribirse a temas" en la página 406.

Otorgar autorización para publicar mensajes en un tema

Otorgue la autorización para publicar mensajes en un tema o un conjunto de temas a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para publicar mensajes en algunos temas, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +pub
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(*PUB) MQMNAME('
QMgrName ')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQTOPIC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para suscribirse a temas
Otorgue la autorización para acceder a un tema o un conjunto de temas a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para suscribirse a algunos temas, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- 

IBM i
- 

Linux
- 

UNIX
- 

Windows

Nota: 

En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +sub
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*TOPIC) USER(GroupName) AUT(*SUB) MQMNAME('
QMgrName ')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQTOPIC QMgrName.SUBSCRIBE.ObjectProfile UACC(NONE)
PERMIT QMgrName.SUBSCRIBE.ObjectProfile CLASS(MQTOPIC) ID(GroupName) ACCESS(UPDATE)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para consultar en un gestor de colas

Otorgue la autorización para consultar en un gestor de colas en cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para consultar en un gestor de colas, utilice los mandatos adecuados de para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName +inq
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*MQM) USER(GroupName) AUT(*INQ) MQMNAME(' QMgrName
')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

  Estos mandatos otorgan acceso al gestor de cola especificado. Para permitir al usuario utilizar el mandato MQINQ, emita los mandatos siguientes:

```
RDEFINE MQCMDS QMgrName.MQINQ.QMGR UACC(NONE)
PERMIT QMgrName.MQINQ.QMGR CLASS(MQCMDS) ID(GroupName) ACCESS(READ)
```

  Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

  ObjectProfile
  El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

  GroupName
  Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para acceder a procesos

Otorgue la autorización para acceder a un proceso o un conjunto de procesos a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para acceder a algunos procesos, utilice los mandatos adecuados de para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i
- Linux
- UNIX
- IBM i

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +all
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(' ObjectProfile ') OBJTYPE(*PRC) USER(GroupName) AUT(*ALL) MQMNAME(' QMgrName
')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQPROC QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile CLASS(MQPROC) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

Otorgar autorización para acceder a listas de nombres

Otorgue la autorización para acceder a una lista de nombres o un conjunto de listas de nombres a cada grupo de usuarios con una necesidad empresarial para ello.

Acerca de esta tarea

Para otorgar autorización para acceder a algunas listas de nombres, utilice los mandatos adecuados para su sistema operativo.

En las plataformas siguientes, también puede utilizar el mandato SET AUTHREC:

- IBM i IBM i
- Linux Linux
- UNIX UNIX
- IBM i Windows

Nota: MQ Appliance En IBM MQ Appliance puede utilizar solo el mandato **SET AUTHREC**.

Procedimiento

- Para sistemas UNIX, Linux, and Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n
ObjectProfile -t namelist -g GroupName
+all
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile
') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALL) MQMNAME('
QMgrName')
```

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQNLIST
QMgrName.ObjectProfile UACC(NONE)
PERMIT QMgrName.ObjectProfile
CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

ObjectProfile
El nombre del objeto o el perfil genérico para el que se van a cambiar autorizaciones.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

ULW Autorización para administrar IBM MQ en UNIX, Linux, and Windows

Los administradores de IBM MQ pueden utilizar todos los mandatos de IBM MQ y otorgar autorizaciones para otros usuarios. Cuando los administradores emiten mandatos a gestores de colas remotos, deben tener la autorización necesaria en el gestor de colas remoto. Se aplican otras Windows.

Los administradores de IBM MQ tienen autorización para utilizar todos los mandatos de IBM MQ (incluidos los mandatos para otorgar autorizaciones de IBM MQ a otros usuarios).

Para ser administrador de IBM MQ, hay que ser miembro de un grupo especial llamado grupo **mqm**.

Windows De forma alternativa, solo en Windows, las cuentas locales pueden administrar IBM MQ si son miembros del grupo de administradores en los sistemas Windows.

Atención: puede añadir el usuario de Azure AD al grupo mqm utilizando un mandato de administrador. Por ejemplo, utilice el mandato `net localgroup mqm AzureAD\<your userID> /add`. A continuación, ejecute los mandatos de administración de IBM MQ o utilice IBM MQ Explorer.

El grupo **mqm** se crea automáticamente cuando se instala IBM MQ. Se pueden añadir más usuarios al grupo para permitirles realizar tareas de administración. Todos los miembros de este grupo tienen acceso a todos los recursos. Este acceso solo se puede revocar eliminando un usuario del grupo **mqm** y ejecutando el mandato **REFRESH SECURITY**.

Los administradores pueden utilizar los mandatos de control para administrar IBM MQ. Uno de estos mandatos de control es **setmqaut**, que se utiliza para conceder autorización a otros usuarios para que puedan acceder a los recursos de IBM MQ o controlarlos. Los mandatos para gestionar registros de autorización PCF están a la disposición de aquellos usuarios que no son administradores a quienes se les han otorgado las autorizaciones dsp y chg en el gestor de colas. Para obtener más información sobre la gestión de las autorizaciones con mandatos PCF, consulte Formatos de mandatos programables.

Los administradores deben tener las autorizaciones necesarias para que el gestor de colas remoto procese los mandatos MQSC. IBM MQ Explorer emite mandatos PCF para realizar tareas de administración. Los administradores no necesitan autorizaciones adicionales para utilizar IBM MQ Explorer para administrar un gestor de colas en el sistema local. Cuando IBM MQ Explorer se utiliza para administrar un gestor de colas en otro sistema, los administradores deben tener las autorizaciones necesarias para que el gestor de colas remoto procese los mandatos PCF.

Atención: En IBM MQ 8.0, no es necesario que sea un administrador para utilizar el mandato de control **runmqsc**, que emite mandatos de script de IBM MQ (MQSC).

Cuando se utiliza **runmqsc** en modalidad indirecta para enviar mandatos MQSC a un gestor de colas remoto, todo mandato MQSC se encapsula en un mandato PCF de escape.

Para obtener más información acerca de las comprobaciones de autorización cuando se procesan los mandatos PCF y MQSC, consulte los temas siguientes:

- Para los mandatos PCF que operan en los gestores de colas, colas, procesos, listas de nombres y objetos de información de autenticación, consulte Autorización para trabajar con objetos de IBM MQ. Consulte en este apartado los mandatos MQSC equivalentes encapsulados en mandatos PCF de escape.
- Para los mandatos PCF que se ejecutan en canales, iniciadores de canal, escuchas y clústeres, consulte Seguridad de canal.
- Para los mandatos PCF que operan en los registros de autorización, consulte Comprobación de autorización para mandatos EN PCF
- z/OS Para los mandatos MQSC que procesa el servidor de mandatos en IBM MQ for z/OS, consulte Seguridad de mandatos y seguridad de recursos de mandatos en z/OS .

Además, en sistemas Windows , la cuenta SYSTEM tiene acceso completo a los recursos de IBM MQ .

En las plataformas UNIX and Linux, también se crea un ID de usuario de **mqm** para uso exclusivo del producto. Este ID no debe estar disponible nunca para los usuarios que no tienen estos privilegios. Todos los objetos de IBM MQ son propiedad del ID de usuario de **mqm**.

En sistemas Windows , los miembros del grupo Administradores también pueden administrar cualquier gestor de colas, al igual que la cuenta SYSTEM. También puede crear un grupo de dominio **mqm** en el controlador de dominios que contenga todos los ID de usuario con privilegios que están activos en el dominio y añadirlo al grupo **mqm** local. Algunos mandatos, por ejemplo **crtmqm**, manipulan autorizaciones sobre objetos de IBM MQ y por ello necesitan autorización para trabajar con estos objetos (tal como se describe en los apartados siguientes). Los miembros del grupo **mqm** tienen autorización para trabajar con todos los objetos, pero en sistemas Windows se puede dar el caso en que se deniegue la autorización si hay un usuario local y un usuario autenticado por el dominio con el mismo nombre. Este tema se describe en el apartado “Principales y grupos en UNIX, Linux, and Windows” en la página 414.

Las versiones de Windows con una característica de Control de cuentas de usuario (UAC) restringe las acciones que los usuarios pueden llevar a cabo en determinados recursos del sistema operativo, incluso si son miembros del grupo Administradores. Si su ID de usuario está en el grupo Administradores pero no en el grupo **mqm**, hay que utilizar un indicador de mandatos elevado para ejecutar mandatos de administración de IBM MQ como, por ejemplo, **crtmqm**; de lo contrario se genera el error `"AMQ7077: No tiene autorización para realizar la operación solicitada"`. Para abrir un indicador de mandatos elevado, pulse el botón derecho del ratón en el elemento de menú, o icono, de inicio, para el indicador de mandatos, y seleccione **Ejecutar como administrador**.

No es necesario ser miembro del grupo **mqm** para realizar las acciones siguientes:

- Emitir mandatos desde un programa de aplicación que emite mandatos PCF, o mandatos MQSC dentro de un mandato PCF de escape, a menos que los mandatos manipulen iniciadores de canal. (Estos mandatos se describen en “Protección de las definiciones de iniciador de canal” en la página 112).
- Emitir llamadas MQI desde un programa de aplicación (a menos que desee utilizar los enlaces de vía rápida en la llamada `MQCONNX`).
- Utilice el mandato `crtmqcvx` para crear un fragmento de código que realice la conversión de datos en estructuras de tipo de datos.
- Utilizar el mandato `dspmq` para visualizar gestores de colas.
- Utilice el mandato `dspmqtrc` para visualizar la salida de rastreo con formato de IBM MQ.

Se aplica una limitación de 12 caracteres al grupo y a los ID de usuario.

Las plataformas UNIX and Linux suelen restringir la longitud de un ID de usuario a 12 caracteres. AIX 5.3 ha aumentado este límite, pero IBM MQ sigue observando una restricción de 12 caracteres en todas las plataformas UNIX and Linux . Si utiliza un ID de usuario de más de 12 caracteres, IBM MQ lo sustituye por el valor UNKNOWN. No defina un ID de usuario con un valor de UNKNOWN.

ULW Gestión del grupo mqm en UNIX, Linux, and Windows

Se otorgan privilegios administrativos completos a los usuarios del grupo mqm a través de IBM MQ. Por este motivo, no debe inscribir aplicaciones y usuarios ordinarios en el grupo mqm. El grupo mqm sólo debe contener las cuentas de los administradores de IBM MQ.

Estas tareas se describen en el apartado:

- Windows Creación y gestión de grupos en Windows
- AIX Creación y gestión de grupos en AIX
- Solaris Creación y gestión de grupos en Solaris
- Linux Creación y gestión de grupos en Linux

Windows Si el controlador de dominio se ejecuta en Windows 2000 o en Windows 2003 o posterior, es posible que el administrador del dominio tenga que establecer una cuenta especial para que la utilice IBM MQ. Para obtener más información, consulte Configuración de IBM MQ con Prepare IBM MQ Wizard y Creación y configuración de cuentas de dominio de Windows para IBM MQ.

ULW Autorización para trabajar con objetos IBM MQ en UNIX, Linux, and Windows

Todos los objetos están protegidos por IBM MQ y los principales deben recibir la autorización adecuada para acceder a ellos. Diferentes principales necesitan diferentes derechos de acceso a diferentes objetos.

Se accede a los gestores de colas, colas, definiciones de proceso, listas de nombres, canales, canales de conexión de cliente, escuchas, servicios y objetos de información de autenticación desde aplicaciones que utilizan llamadas MQI o mandatos PCF. Estos recursos están todos protegidos por IBM MQ, y las aplicaciones deben tener permiso para acceder a ellos. La entidad que realiza la solicitud puede ser un usuario, un programa de aplicación que emite una llamada MQI o un programa de administración que emite un mandato PCF. Se hace referencia al identificador del peticionario como *principal*.

Se puede otorgar a distintos grupos de principales diferentes tipos de autorización de acceso al mismo objeto. Por ejemplo, para una cola específica, puede permitirse a un grupo que realice operaciones de transferir y obtener; otro grupo puede tener únicamente autorización para examinar la cola (MQGET con la opción de examinar). Del mismo modo, algunos grupos pueden tener autorización de transferir y obtener para una cola pero pueden no tener autorización para alterar los atributos de la cola o suprimirla.

Algunas operaciones son especialmente comprometidas y deberían limitarse a usuarios con privilegios. Por ejemplo:

- Acceder a algunas colas especiales, tales como las colas de transmisión o la cola de mandatos SYSTEM.ADMIN.COMMAND.QUEUE
- La ejecución de programas que utilicen todas las opciones de contexto de la MQI
- La creación y supresión de colas de aplicación

Se concede automáticamente permiso de acceso completo sobre un objeto al ID de usuario que ha creado el objeto y a todos los miembros del grupo mqm (y también a los miembros del grupo Administradores en sistemas Windows).

Conceptos relacionados

“Autorización para administrar IBM MQ en UNIX, Linux, and Windows” en la página 409

Los administradores de IBM MQ pueden utilizar todos los mandatos de IBM MQ y otorgar autorizaciones para otros usuarios. Cuando los administradores emiten mandatos a gestores de colas remotos, deben tener la autorización necesaria en el gestor de colas remoto. Se aplican otras Windows.

ULW Cuándo se realizan comprobaciones de seguridad en UNIX, Linux, and Windows

Las comprobaciones de seguridad normalmente se realizan al conectar a un gestor de colas, al abrir o cerrar objetos y al transferir u obtener mensajes.

Las comprobaciones de seguridad que se realizan en una aplicación típica son las siguientes:

Conectar al gestor de colas (llamadas MQCONN o MQCONNX)
Ésta es la primera vez que la aplicación se asocia a un gestor de colas determinado. El gestor de colas investiga en el entorno operativo para detectar el ID de usuario asociado a la aplicación. A continuación, IBM MQ comprueba que el ID de usuario tiene autorización para conectarse con el gestor de colas y guarda el ID de usuario para futuras comprobaciones.

Los usuarios no necesitan iniciar la sesión en IBM MQ; IBM MQ presupone que los usuarios han iniciado la sesión en el sistema operativo y que éste los ha autenticado.

Abrir el objeto (llamadas MQOPEN o MQPUT1)
Se accede a los objetos IBM MQ abriendo el objeto y emitiendo mandatos para el mismo. Todas las comprobaciones de recursos se realizan cuando se abre el objeto, en lugar de hacerlo cuando se accede al mismo. Esto significa que la solicitud **MQOPEN** debe especificar el tipo de acceso necesario (por ejemplo, si el usuario simplemente desea examinar el objeto o realizar una actualización como, por ejemplo, colocar mensajes en una cola).

IBM MQ comprueba el recurso que se nombra en la solicitud **MQOPEN**. En un objeto de cola alias o remota, la autorización que se utiliza es la del objeto propiamente dicho, no la de la cola en la que se resuelve la cola alias o remota. Esto significa que el usuario no necesita tener permiso para acceder al mismo. Limite la autorización para crear colas a los usuarios con privilegios. De otro modo, los usuarios podrán eludir el control de accesos normal simplemente creando un alias. Si se hace referencia a una cola remota de forma explícita en los nombres de la cola y del gestor de colas, se comprobará la cola de transmisión asociada al gestor de colas remoto.

La autorización sobre una cola dinámica se basa en la cola modelo de la que se deriva, aunque no tiene por qué ser igual. Este tema se describe en la Nota “1” en la página 132.

El ID de usuario que utiliza el gestor de colas para las comprobaciones de acceso es el ID de usuario obtenido desde el sistema operativo de la aplicación conectada al gestor de colas. Una aplicación con las autorizaciones adecuadas puede emitir una llamada **MQOPEN** especificando un ID de usuario alternativo. Las comprobaciones de control de accesos se realizan de este modo en el ID de usuario alternativo. Esto no modifica el ID de usuario asociado a la aplicación, solamente el que se utiliza para las comprobaciones de control de accesos.

Transferir y obtener mensajes (llamadas MQPUT o MQGET)
No se realizan comprobaciones de control de acceso.

Cerrar el objeto (MQCLOSE)
No se realizan comprobaciones de control de accesos a menos que el resultado de la llamada **MQCLOSE** sea la supresión de una cola dinámica. En este caso, se comprueba que el ID de usuario tenga autorización para suprimir la cola.

Suscripción a un tema (MQSUB)
Cuando una aplicación se suscribe a un tema, especifica el tipo de operación que necesita realizar. La operación es crear una nueva suscripción, alterar una suscripción existente o reanudar una suscripción existente sin modificarla. Para cada tipo de operación, el gestor de colas comprueba que el ID de usuario asociado a la aplicación tenga autorización para realizar la operación.

Cuando una aplicación se suscribe a un tema, las comprobaciones de autorización se realizan respecto a objetos de temas que se encuentran en el árbol de temas en el punto o por encima

del punto del árbol de temas al que se ha suscrito la aplicación. Las comprobaciones de autorización pueden implicar comprobaciones en más de un objeto de tema.

El ID de usuario que utiliza el gestor de colas para las comprobaciones de autorización es el ID de usuario que ha obtenido del sistema operativo cuando la aplicación se conecta al gestor de colas.

El gestor de colas realiza comprobaciones de autorización en las colas de suscriptores pero no en las colas gestionadas.

ULW Cómo implementa IBM MQ el control de accesos en UNIX, Linux, and Windows

IBM MQ utiliza los servicios de seguridad proporcionados por el sistema operativo subyacente mediante el gestor de autorizaciones sobre objetos. IBM MQ proporciona mandatos para crear y mantener listas de control de accesos.

Una interfaz de control de accesos llamada Interfaz del servicio de autorización forma parte de IBM MQ. IBM MQ proporciona una implementación de un gestor de control de accesos (conforme a la Interfaz del servicio de autorización) que se conoce como *gestor de autorizaciones sobre objetos (OAM)*. Este gestor se instala y se activa automáticamente para cada gestor de colas que cree, a menos que especifique lo contrario, como se explica en "Impedir comprobaciones de acceso de seguridad en los sistemas UNIX, Linux, and Windows" en la página 367). El OAM puede sustituirse por cualquier componente escrito por el usuario o por terceros que esté en conformidad con la Interfaz del servicio de autorización.

El OAM aprovecha las características de seguridad del sistema operativo subyacente, utilizando los ID de usuario y de grupo del sistema operativo. Los usuarios sólo pueden acceder a los objetos de IBM MQ si tienen la autorización correcta. En el apartado "Control del acceso a objetos mediante el OAM en UNIX, Linux, and Windows" en la página 357 se explica cómo conceder y denegar esta autorización.

El OAM mantiene una ACL (Access Control List - Lista de control de accesos) para cada recurso que controla. Los datos de autorización se almacenan en una cola local llamada SYSTEM.AUTH.DATA.QUEUE. El acceso a esta cola está restringido a los usuarios del grupo mqm, y adicionalmente en Windows, a los usuarios del grupo Administradores y a los usuarios que han iniciado sesión con el ID SYSTEM. El acceso de usuarios a la cola no se puede cambiar.

IBM MQ proporciona mandatos para crear y mantener listas de control de accesos. Para obtener más información acerca de estos mandatos, consulte "Control del acceso a objetos mediante el OAM en UNIX, Linux, and Windows" en la página 357.

IBM MQ pasa al OAM una solicitud que contiene un principal, un nombre de recurso y un tipo de acceso. El OAM otorga o deniega el acceso basándose en la ACL que mantiene. IBM MQ sigue la decisión adoptada por el OAM; si el OAM no puede tomar una decisión, IBM MQ no permite el acceso.

ULW Identificación del ID de usuario en UNIX, Linux, and Windows

El gestor de autorizaciones sobre objetos identifica el principal que está solicitando acceso a un recurso. El ID de usuario utilizado como principal varía según el contexto.

El gestor de autorizaciones sobre objetos (OAM) debe poder identificar quién solicita acceso a un recurso determinado. IBM MQ utiliza el término *principal* para referirse a este identificador. El principal se establece cuando la aplicación se conecta por primera vez al gestor de colas; lo determina el gestor de colas a partir del ID de usuario asociado a la aplicación de conexión. (Si la aplicación emite llamadas XA sin establecer conexión con el gestor de colas, el ID de usuario asociado con la aplicación que emite la llamada xa_open se utiliza para las comprobaciones de autorizaciones que realiza el gestor de colas.)

En sistemas UNIX and Linux, las rutinas de autorización comprueban el ID de usuario real (conectado) o el ID de usuario efectivo asociado a la aplicación. El ID de usuario comprobado puede depender del tipo de enlace; para obtener información detallada, consulte Servicios instalables.

IBM MQ propaga el ID de usuario que recibe del sistema en la cabecera de mensaje (la estructura MQMD) de cada mensaje para identificar al usuario. Este identificador forma parte de la información de contexto del mensaje y se describe en el apartado "Autorización de contexto en UNIX, Linux, and Windows" en la

página 416. Las aplicaciones no pueden alterar esta información a menos que tengan autorización para cambiar la información de contexto.

ULW *Principales y grupos en UNIX, Linux, and Windows*

Los principales pueden pertenecer a grupos. Al otorgar el acceso a recursos a grupos en vez de a usuarios individuales, puede reducir la cantidad de administración necesaria. Las listas de control de acceso (ACL) se basan en grupos y en los ID de usuario.

Por ejemplo, puede definir un grupo que conste de usuarios que deseen ejecutar una aplicación determinada. A otros usuarios se les puede permitir el acceso a todos los recursos que necesiten añadiendo su ID de usuario al grupo adecuado.

Este proceso de definir y gestionar grupos se describe para plataformas concretas:

- Windows Creación y gestión de grupos en Windows
- AIX Creación y gestión de grupos en AIX
- Solaris Creación y gestión de grupos en Solaris
- Linux Creación y gestión de grupos en Linux

Un principal puede pertenecer a más de un grupo (su conjunto de grupos). Tiene la suma de todas las autorizaciones que se han otorgado a cada grupos en su conjunto de grupos. Estas autorizaciones se almacenan en memoria caché, de este modo los cambios que realice en la pertenencia del grupo del principal no se reconocen, hasta que se reinicia el gestor de colas, a menos que emita el mandato MQSC **REFRESH SECURITY** (o su PCF equivalente).

Linux UNIX Sistemas UNIX and Linux

A partir de IBM MQ 8.0, las listas de control de accesos (ACL) se basan en los ID de usuario y, también, en grupos y puede utilizar la autorización estableciendo el atributo **SecurityPolicy** en el valor apropiado, tal como se describe en Configuración de servicios instalables y Configuración de stanzas de servicio de autorización en UNIX y Linux.

Desde IBM MQ 8.0, puede utilizar el *modelo basado en usuario* para la autorización y esto le permite utilizar tanto usuarios como grupos. Sin embargo, al especificar un usuario en el mandato setmqaut, los nuevos permisos se aplican solo a dicho usuario y no a los grupos a los que pertenece dicho usuario. Si desea más información, consulte Permisos basados en usuarios OAM en sistemas UNIX y Linux.

Cuando se utiliza el *modelo basado en grupo* para la autorización, el grupo primario al que pertenece el ID de usuario se incluye en la ACL. El ID de usuario individual no se incluye y la autorización se otorga a todos los miembros de dicho grupo. Por eso, tenga en cuenta que puede modificar accidentalmente la autorización de un principal al modificar la autorización de otro principal del mismo grupo.

Todos los usuarios se asignan nominalmente al grupo de usuarios predeterminado `nadie` y de forma predeterminada, a este grupo no se le concede ningún tipo de autorización. Puede cambiar la autorización del grupo `nadie` para otorgar acceso a los recursos IBM MQ a aquellos usuarios que no tienen autorizaciones específicas.

No defina un ID de usuario con el valor `UNKNOWN`. El valor `UNKNOWN` se utiliza cuando ID de usuario es demasiado largo, por lo que los ID de usuario arbitrarios deberían utilizar las autorizaciones de acceso de `UNKNOWN`.

Los ID de usuario tener 12 caracteres como máximo y los nombres de grupo también.

Windows Sistemas Windows

Las ACL están basadas en los grupos y en los ID de usuario. Las comprobaciones son las mismas que para UNIX. Puede tener usuarios diferentes en dominios diferentes con el mismo ID de usuario. IBM MQ permite que los ID de usuario se califiquen con el nombre de dominio, de modo que estos usuarios puedan tener diferentes niveles de acceso.

El nombre del grupo puede incluir opcionalmente un nombre de dominio, especificado con los formatos siguientes:

```
GroupName@domain domain_name\group_name
```

Los grupos globales son comprobados por el OAM sólo en dos casos:

1. La stanza de seguridad del gestor de colas incluye el valor: `GroupModel=GlobalGroups`. Consulte Seguridad.
2. El gestor de colas está utilizando un grupo de acceso de seguridad alternativo. Consulte **crtmqm**.

Los ID de usuario pueden tener hasta 20 caracteres, los nombres de dominio hasta 15 caracteres y los nombres de grupo hasta 64 caracteres.

El OAM comprueba en primer lugar la base de datos de seguridad local, luego la base de datos del dominio primario y, finalmente, la base de datos de cualquier dominio fiable. Para la comprobación, el OAM utiliza el primer ID de usuario que encuentra. Cada uno de estos ID de usuario puede tener distintos miembros de grupos en un sistema determinado.

Algunos mandatos de control (por ejemplo, **crtmqm**) modifican las autorizaciones sobre objetos IBM MQ utilizando el gestor de autorizaciones sobre objetos (OAM). El OAM busca en las bases de datos de seguridad en el orden dado para determinar los derechos de autorización de un ID de usuario específico. La autorización que determine el OAM puede alterar temporalmente el que un ID de usuario sea miembro del grupo mqm local. Por ejemplo, si emite el mandato **crtmqm** desde un ID de usuario autenticado por un controlador de dominio que sea miembro del grupo mqm local a través de un grupo global, el mandato no se ejecutará correctamente si el sistema tiene un usuario local con el mismo nombre que no esté en el grupo mqm local.

Si desea más información sobre cómo establecer el atributo **SecurityPolicy** en Windows, consulte Servicios instalables y Configuración de stanzas de servicio de autorización en Windows.

Windows *Identificadores de seguridad (SID) de Windows*

IBM MQ en Windows utiliza el identificador de seguridad (SID) cuando está disponible. Si no se proporciona un SID de Windows con una solicitud de autorización, IBM MQ identifica el usuario basándose solamente en el nombre de usuario, pero esto puede dar como resultado que se otorgue la autorización incorrecta.

En sistemas Windows, el identificador de seguridad (SID) se utiliza para complementar el ID de usuario. El SID contiene información que identifica todos los detalles de la cuenta del usuario en la base de datos del administrador de cuentas de seguridad (SAM) de Windows donde se ha definido el usuario. Cuando se crea un mensaje en IBM MQ for Windows, IBM MQ almacena el SID en el descriptor de mensaje. Cuando IBM MQ en Windows realiza comprobaciones de autorización, utiliza el SID para consultar la información completa en la base de datos SAM. Para que esta consulta se lleve a cabo correctamente, la base de datos SAM en la que está definido el usuario debe estar accesible.

De forma predeterminada, si no se proporciona un SID de Windows con una solicitud de autorización, IBM MQ identifica el usuario basándose solamente en el nombre de usuario. Esto se efectúa buscando en las bases de datos de seguridad en el orden siguiente:

1. La base de datos de seguridad local.
2. La base de datos de seguridad del dominio primario.
3. La base de datos de seguridad de dominios fiables.

Si el nombre de usuario no es exclusivo, se puede otorgar una autorización IBM MQ incorrecta. Para evitar este problema, incluya un SID en cada solicitud de autorización; IBM MQ utiliza el SID para establecer las credenciales de usuario.

Para especificar que todas las peticiones de autorización deben incluir un SID, utilice **regedit**. Establezca SecurityPolicy en NTSIDsRequired.

ULW Autorización de usuario alternativo en UNIX, Linux, and Windows

Puede especificar que un ID de usuario puede utilizar la autorización de otro usuario cuando accede a un IBM MQ. Esto se denomina *autorización de usuario alternativo* y puede utilizarla en cualquier objeto IBM MQ MQ.

La autorización de usuario alternativo es esencial cuando un servidor recibe peticiones de un programa y desea asegurarse de que el programa tiene la autorización necesaria para la solicitud. El servidor puede tener la autorización necesaria, pero necesita saber si el programa tiene autorización para las acciones que ha solicitado.

Por ejemplo, suponga que un programa servidor que se está ejecutando bajo el ID PAYSERV recupera de una cola un mensaje de solicitud que había transferido a la cola el ID de usuario USER1. Cuando el programa servidor obtiene el mensaje de solicitud, procesa la solicitud y vuelve a transferir la respuesta a la cola de respuestas especificada con el mensaje de solicitud. En lugar de utilizar su propio ID de usuario (PAYSERV) para autorizar la apertura de una cola de respuestas, el servidor puede especificar otro ID de usuario, en este caso, USER1. En este ejemplo, puede utilizar la autorización de usuario alternativo para controlar si PAYSERV puede especificar USER1 como ID de usuario alternativo al abrir la cola de respuestas.

El ID de usuario alternativo se especifica en el campo **AlternateUserId** del descriptor de objeto.

ULW Autorización de contexto en UNIX, Linux, and Windows

El contexto es la información que se aplica a un mensaje determinado y está contenida en el descriptor de mensaje, MQMD, que forma parte del mensaje. Las aplicaciones pueden especificar los datos de contexto cuando se realiza una llamada MQOPEN o MQPUT.

En la información de contexto hay dos secciones:

Sección de identidad
De quién procede el mensaje. Consta de los campos UserIdentifier, AccountingTokeny ApplIdentityData .

Sección de origen
De dónde procede el mensaje y cuándo se ha transferido a la cola. Consta de los campos PutApplType, PutApplName, PutDate, PutTimey ApplOriginData .

Las aplicaciones pueden especificar los datos de contexto cuando se realiza una llamada MQOPEN o MQPUT. Estos datos pueden haber sido generados por la aplicación, transmitidos desde otro mensaje o generados por el gestor de colas predeterminado. Por ejemplo, los programas servidor pueden utilizar los datos de contexto para comprobar la identidad del peticionario, con lo que comprueban si el mensaje procede de una aplicación que se ejecuta bajo un ID de usuario autorizado.

Un programa servidor puede utilizar el campo UserIdentifier para determinar el ID de usuario de un usuario alternativo. La autorización de contexto se utiliza para controlar si el usuario puede especificar alguna de las opciones de contexto en cualquier MQOPEN o MQPUT1.

Consulte Control de la información de contexto para obtener información sobre las opciones de contexto, y Visión general de MQMD para obtener las descripciones de los campos del descriptor de mensaje relativos al contexto.

Implementación de control de accesos en salidas de seguridad

Puede implementar control de accesos en una salida de seguridad utilizando el campo MCAUserIdentifier o el gestor de autorizaciones sobre objetos.

MCAUserIdentifier

Cada instancia de un canal actual tiene una estructura de definición de canal, MQCD, asociada. Los valores iniciales de los campos de MQCD se determinan mediante la definición de canal que crea un administrador de IBM MQ. En particular, el valor inicial de uno de los campos, *MCAUserIdentifier* , se

determina mediante el valor del parámetro MCAUSER del mandato DEFINE CHANNEL o mediante el equivalente a MCAUSER si la definición de canal se crea de otra forma.

La estructura MQCD se pasa a un programa de salida de canal al que llama un MCA. Cuando un MCA llama a una salida de seguridad, la salida de seguridad puede cambiar el valor de *MCAUserIdentifier*, sustituyendo el valor especificado en la definición de canal.

Multi En Multiplatforms, a menos que el valor de *MCAUserIdentifier* esté en blanco, el gestor de colas utiliza el valor de *MCAUserIdentifier* como ID de usuario para las comprobaciones de autorización cuando un MCA intenta acceder a los recursos del gestor de colas después de que se haya conectado al gestor de colas. Si el valor de *MCAUserIdentifier* está en blanco, el gestor de colas utiliza en su lugar el ID de usuario predeterminado del MCA. Esto es aplicable a los canales RCVR, RQSTR, CLUSRCVR y SVRCONN. Para los MCA emisores, el ID de usuario predeterminado se utiliza siempre para las comprobaciones de autorización, incluso si el valor de *MCAUserIdentifier* no está en blanco.

z/OS En z/OS, el gestor de colas puede utilizar el valor de *MCAUserIdentifier* para comprobaciones de autorización, siempre y cuando no esté en blanco. Para los MCA receptores y los MCA de conexión con servidor, el hecho de que el gestor de colas utilice el valor de *MCAUserIdentifier* para comprobaciones de autoridad depende de:

- El valor del parámetro PUTAUT en la definición de canal
- El perfil RACF utilizado para las comprobaciones
- El nivel de acceso del ID de usuario del espacio de direcciones del iniciador de canal ante el perfil RESLEVEL

Para los MCA emisores, depende de:

- Si el MCA emisor efectúa la llamada o envía la respuesta
- El nivel de acceso del ID de usuario del espacio de direcciones del iniciador de canal ante el perfil RESLEVEL

El ID de usuario que una salida de seguridad almacena en *MCAUserIdentifier* se puede adquirir de varias formas. A continuación, se detallan algunos ejemplos:

- Suponiendo que no hay ninguna salida de seguridad en el extremo del cliente de un canal MQI, un ID de usuario asociado con la aplicación cliente IBM MQ fluye desde el MCA de conexión del cliente al MCA de conexión del servidor cuando la aplicación cliente emite una llamada MQCONN.El MCA de conexión del servidor almacena su ID de usuario en el campo *RemoteUserIdentifier* de la estructura de definición de canal, MQCD. Si el valor de *MCAUserIdentifier* está en blanco en este momento, el MCA almacena el mismo ID de usuario en *MCAUserIdentifier*. Si el MCA no almacena el ID de usuario en *MCAUserIdentifier*, una salida de seguridad puede hacerlo posteriormente, estableciendo *MCAUserIdentifier* en el valor de *RemoteUserIdentifier*.

  Si el ID de usuario que fluye del sistema cliente está entrando en un nuevo dominio de seguridad y no es válido en el sistema servidor, la salida de seguridad puede sustituir el ID de usuario por uno que sea válido y almacenar el ID de usuario sustituido en *MCAUserIdentifier*.
- La salida de seguridad del asociado puede enviar el ID de usuario en un mensaje de seguridad.

  En un canal de mensaje, una salida de seguridad a la que ha llamado el MCA emisor puede enviar el ID de usuario bajo el cual se ejecuta el MCA emisor. Luego, una salida de seguridad a la que ha llamado el MCA receptor puede almacenar el ID de usuario en *MCAUserIdentifier* . De forma similar, en un canal MQI, una salida de seguridad en el extremo del cliente del canal puede enviar el ID de usuario asociado con la aplicación IBM MQ MQI client. Luego una salida de seguridad en el extremo del servidor del canal puede almacenar el ID de usuario en *MCAUserIdentifier*. Como en el ejemplo anterior, si el ID de usuario no es válido en el sistema de destino, la salida de seguridad puede sustituir el ID de usuario por uno que sea válido y almacenar el ID de usuario sustituido en *MCAUserIdentifier*.

  Si se recibe un certificado digital como parte del servicio de identificación y autenticación, una salida de seguridad puede correlacionar el Nombre distinguido del certificado con un ID de usuario que sea válido en el sistema de destino. Puede almacenar el ID de usuario en *MCAUserIdentifier*.

- Si TLS se utiliza en el canal, el nombre distinguido del asociado (DN) se pasa a la salida del campo SSLPeerNamePtr de MQCD y el DN de emisor del certificado se pasa a la salida del campo SSLRemCertIssNamePtr de MQCXP.

Para obtener más información sobre el campo *MCAUserIdentifier*, la estructura de definición de canal, MQCD, y la estructura del parámetro de salida de canal, MQCXP, consulte Llamadas de salida de canal y estructuras de datos. Para obtener más información sobre el ID de usuario que fluye desde un sistema cliente en un canal MQI, consulte Control de accesos.

Nota: Las aplicaciones de salida de seguridad creadas antes del release de IBM WebSphere MQ 7.1 pueden requerir actualización. Para obtener más información, consulte Programas de salida de seguridad de canal.

Autenticación de usuarios del gestor de autorizaciones sobre objetos de IBM MQ

En las conexiones de cliente MQI de IBM MQ MQI client, las salidas de seguridad se pueden utilizar para modificar o crear la estructura MQCSP utilizada en una autenticación de usuario del gestor de autorizaciones sobre objetos (OAM). Esto se describe en la sección Programas de salida de canal para canales de mensajería

Implementación de control de accesos en salidas de mensajes

Es posible que tenga que utilizar una salida de mensajes para sustituir un ID de usuario por otro.

Considere una aplicación cliente que envía un mensaje a una aplicación de servidor. La aplicación de servidor puede extraer el ID de usuario del campo *UserIdentifier* del descriptor de mensaje y, siempre y cuando tenga autorización de usuario alternativo, solicitar al gestor de colas que utilice este ID de usuario para comprobaciones de autorización cuando acceda a recursos de IBM MQ en nombre del cliente.

Si el parámetro PUTAUT tiene el valor CTX (o ALTMCA en z/OS) en la definición de canal, el ID de usuario del campo *UserIdentifier* de cada mensaje de entrada se utiliza para comprobaciones de autorización cuando el MCA abre la cola de destino.

En determinadas circunstancias, cuando se genera un mensaje de informe, este se coloca utilizando la autoridad del ID de usuario del campo *UserIdentifier* del mensaje que ha originado el informe. En particular, los informes de confirmación de entrega (COD) y los informes de caducidad siempre se colocan con esta autoridad.

Debido a estas situaciones, es posible que sea necesario sustituir un ID de usuario por otro en el campo *UserIdentifier* cuando un mensaje entra en un nuevo dominio de seguridad. Esto se puede hacer mediante una salida de mensajes en el extremo receptor del canal. Como alternativa, puede asegurarse de que el ID de usuario del campo *UserIdentifier* de un mensaje de entrada está definido en el nuevo dominio de seguridad.

Si un mensaje de entrada contiene un certificado digital correspondiente al usuario de la aplicación que ha enviado el mensaje, una salida de mensajes puede validar el certificado y correlacionar el Nombre distinguido del certificado con un ID de usuario que sea válido en el sistema receptor. Luego puede establecer el campo *UserIdentifier* del descriptor de mensajes en este ID de usuario.

Si es necesario que una salida de mensajes cambie el valor del campo *UserIdentifier* en un mensaje de entrada, es posible que la salida de mensajes tenga que autenticar el emisor del mensaje al mismo tiempo. Para obtener más información, consulte “Correlación de identidad en salidas de mensajes” en la página 341.

Implementación de control de accesos en la salida de API y la salida cruzada de API

Una salida de API o una salida cruzada de API puede proporcionar controles de accesos para complementar los que proporciona IBM MQ. En concreto, la salida puede proporcionar control de accesos a nivel de mensaje. La salida puede garantizar que una aplicación transfiera a una cola, u obtenga de una cola, sólo aquellos mensajes que cumplen ciertos criterios.

Tenga en cuenta los ejemplos siguientes:

- Un mensaje contiene información sobre un pedido. Cuando una aplicación intenta transferir un mensaje a una cola, una salida de API o salida cruzada de API puede comprobar si el valor total del pedido es inferior a un límite establecido previamente.
- Llegan mensajes a una cola de destino desde gestores de colas remotos. Cuando una aplicación intenta obtener un mensaje de la cola, una salida de API o salida cruzada de API puede comprobar si el emisor del mensaje tiene autorización para enviar un mensaje a la cola.

Autorización LDAP

Puede utilizar la autorización LDAP para eliminar la necesidad de un ID de usuario local.

Disponibilidad de la autorización LDAP en plataformas soportadas

La autorización LDAP está disponible en las siguientes plataformas:

- UNIX
- IBM i
- Windows

Atención:

Desde la disponibilidad general de IBM MQ 9.0, esta funcionalidad está disponible en todos los gestores de colas, ya sean nuevos o migrados de un release anterior.

Visión general de la autorización LDAP

Con la autorización LDAP, los mandatos que manejan la configuración de autorización, como por ejemplo **setmqaut** y **DISPLAY AUTHREC**, pueden procesar nombres distinguidos. Anteriormente, los usuarios se autenticaban comparando sus credenciales con el máximo de caracteres disponibles que existen para los usuarios y grupos en el sistema operativo local.

Atención: Si ha ejecutado el mandato **DEFINE AUTHINFO**, debe reiniciar el gestor de colas. Si no se reinicia el gestor de colas, el mandato setmqaut no devuelve el resultado correcto.

Si un usuario proporciona un ID de usuario en lugar de un nombre distinguido, el ID de usuario se procesa. Por ejemplo, cuando haya un mensaje entrante en un canal con PUTAUT(CTX), los caracteres en el ID de usuario se correlacionan con un nombre distinguido LDAP y se realizan las comprobaciones de autorización correspondientes.

Otros mandatos como **DISPLAY CONN**, siguen trabajando con el ID de usuario y muestran el valor real para el ID de usuario, aunque este ID de usuario pueda no existir realmente en el sistema operativo local.

UNIX Cuando la autorización LDAP se aplica, el gestor de colas siempre utiliza el modelo de usuario de seguridad en plataformas UNIX, independientemente del atributo **SecurityPolicy** en el archivo `qm.ini`. Por lo tanto, el establecimiento de permisos para un usuario individual afecta solo a dicho usuario y no afectará a ningún otro que pertenezca a cualquiera de los grupos de dicho usuario.

Al igual que con el modelo del sistema operativo, un usuario aún tiene la autorización combinada que se ha asignado tanto a las personas como a los grupos (si hay alguno) a los que pertenece el usuario.

Por ejemplo, suponga que se han definido los siguientes registros en un repositorio LDAP.

- En la clase **inetOrgPerson**:

```
dn="cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
        email=JohnDoe1@yourcompany.com [longer than 12 characters]
        shortu=jodoe
        Phone=1234567
```

- En la clase **groupOfNames**:

```
dn="cn=Application Group A, ou=groups, o=yourcompany, c=yourcountry"
        longname=ApplicationGroupA [longer than 12 characters]
        members="cn=JaneDoe, ou=users, o=yourcompany, c=yourcountry",
                "cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry"
```

A efectos de autenticación, se debe haber definido un gestor de colas que utiliza este servidor LDAP de modo que su valor **CONNAUTH** apunte a un objeto **AUTHINFO** del tipo IDPWLDAP, y cuyos atributos de resolución de nombres relevantes probablemente se establezcan de la forma siguiente:

```
USRFIELD(email) SHORTUSR(shortu)
BASEDNU(ou=users,o=yourcompany,c=yourcountry) CLASSUSR(inetOrgPerson)
```

Con esta configuración de autenticación, una aplicación puede completar el campo CSPUserID, que se utiliza dentro de la llamada MQCNO, con uno de los siguientes conjuntos de valores:

```
" cn=JohnDoe ", " JohnDoe1@yourcompany.com ", " email=JohnDoe1@yourcompany.com "
```

o

```
" cn=JohnDoe, ou=users, o=ibm, c=uk ", " shortu=jodoe "
```

En cualquiera de los casos, el sistema puede utilizar los valores proporcionados para autenticar el contexto de sistema operativo de "jodoe".

Configuración de autorizaciones

Cómo utilizar el nombre abreviado o **USRFIELD** para definir autorizaciones.

El método de trabajar con varios formatos, que se describe en “Autorización LDAP” en la página 419, continúa en los mandatos de autorización, con una extensión adicional que el shortname o el USRFIELD se pueden utilizar de forma no adornada.

La serie de caracteres especifica un atributo concreto en el registro LDAP cuando se especifican usuarios (principales) para la autorización.

Importante: La serie de caracteres no debe contener el carácter =, porque este carácter no se puede utilizar en un ID de usuario del sistema operativo.

Si pasa un nombre principal al OAM para la autorización que es potencialmente un shortname, la serie de caracteres debe caber en 12 caracteres. El algoritmo de correlación primero intenta resolverlo en un DN utilizando el atributo SHORTUSR en su consulta LDAP.

Si esto falla con un error UNKNOWN_ENTITY, o si la serie dada no puede ser posiblemente un shortname, se realiza un intento adicional utilizando el atributo USRFIELD para construir la consulta LDAP.

Atención: Si se ha ejecutado el mandato DEFINE AUTHINFO, hay que reiniciar el gestor de colas. Si no se reinicia el gestor de colas, el mandato setmqaut no devuelve el resultado correcto.

A la hora de procesar las autorizaciones de usuario, los siguientes valores del mandato setmqaut son equivalentes.

Tabla 72. Valores de autorización de usuario

Mandato	Nota
`setmqaut -m QM -t qmgr -p jodoe +connect`	Es un nombre plano no calificado, resuelto a través de SHORTUSR.

Tabla 72. Valores de autorización de usuario (continuación)	
Mandato	**Nota**
`setmqaut -m QM -t qmgr -pJohnDoe1@yourcompany.com +connect`	También un nombre plano no calificado, que se resuelve a través de USRFIELD en la misma entidad,
`setmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com +connect`	Utilizando un atributo especificado.
`setmqaut -m QM -t qmgr -p "phone=1234567" +connect`	Utilizando otro atributo especificado que no tiene que ser ninguno de los configurados en el objeto AUTHINFO.

Puede utilizar el mandato SET AUTHREC MQSC como alternativa al mandato **setmqaut**:

```
SET AUTHREC OBJTYPE(QMGR) PRINCIPAL('JohnDoe1@yourcompany.com') AUTHADD(connect)
```

o el mandato PCF Establecer registro de autorización (MQCMD_SET_AUTH_REC) con el elemento MQCACF_PRINCIPAL_ENTITY_NAMES que contiene la serie:

```
 "cn=JohnDoe,ou=users,o=yourcompany,c=yourcountry"
```

Al procesar grupos, no hay ambigüedad sobre el proceso de `shortname` , ya que no hay ningún requisito para ajustar cualquier forma de un nombre de grupo en 12 caracteres. Por lo tanto, no existe ningún equivalente del atributo SHORTUSR para grupos.

Esto significa que los ejemplos de sintaxis descritos en Tabla 73 en la página 421 son válidos suponiendo que ha se configurado el objeto AUTHINFO con los atributos ampliados y se ha establecido en:

```
GRPFIELD(longname)
BASEDNG(ou=groups,o=yourcompany,c=yourcountry ) CLASSGRP(groupOfNames)
```

Tabla 73. Valores de autorización de grupo	
Mandato	**Nota**
`setmqaut -m QM -t qmgr -g ApplicationGroupA +connect`	Utilizando GRPFIELD para resolver
`setmqaut -m QM -t qmgr -g longname=ApplicationGroupA +connect`	Especificando un solo atributo
`setmqaut -m QM -t qmgr -g "cn=Application Group A,ou=groups,o=yourcompany,c=yourcountry" +connect`	Utilizando el nombre distinguido completo

Puede utilizar el mandato MQSC SET AUTHREC como alternativa al mandato **setmqaut** anterior:

```
SET AUTHREC OBJTYPE(QMGR) GROUP('ApplicationGroupA')
      AUTHADD(connect)
```

o el mandato PCF Establecer registro de autorización (MQCMD_SET_AUTH_REC) con el elemento MQCACF_GROUP_ENTITY_NAMES que contiene la serie:

```
 "ApplicationGroupA"
```

Importante:

Sea cual sea el formato que se utiliza para hacer referencia a un nombre, ya sea para usuario o grupo, debe ser posible obtener un DN exclusivo.

Así, por ejemplo, no debe tener dos registros distintos que sean "`shortu=jodoe`".

Si no puede determinarse un solo nombre distinguido exclusivo, el OAM devuelve MQRC_UNKNOWN_ENTITY.

Visualización de autorizaciones

Hay distintos métodos de visualizar autorizaciones de usuarios o grupos.

Mandato dspmqaut

El método más simple para visualizar las autorizaciones que está disponible para un usuario o grupo es utilizar el mandato dspmqaut.

Puede utilizar una consulta en cualquiera de las variaciones de sintaxis para identificar un usuario o un grupo. Tenga en cuenta que la salida del mandato repite la identidad en el formato especificado en la línea de mandatos. La salida no informa sobre el nombre distinguido completo resuelto.

Por ejemplo:

```
dspmqaut -m QM -t qmgr -p johndoe
Entity johndoe has the following authorizations for object QM:
    connect
```

o

```
dspmqaut -m QM -t qmgr -p email=JohnDoe1@yourcompany.com
Entity email=JohnDoe1@yourcompany.com has the following authorizations for object QM:
    connect
```

Mandatos dmpmqaut y dmpmqcfg

El mandato dmpmqaut y sus mandatos MQSC o PCF equivalentes, puede especificar el principal o el grupo en cualquiera de los formatos soportados, como las tablas de **setmqaut** descritas en “Configuración de autorizaciones” en la página 420. Sin embargo, a diferencia de **dspmqaut**, el mandato **dmpmqaut** siempre notifica el nombre distinguido completo.

```
dmpmqaut -m QM -t qmgr -p jodoe
------------------------------------
profile: self
object type:qmgr
entity:cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry
entity type: principal
authority: connect
```

Del mismo modo, el mandato dmpmqcfg, que no tiene ningún filtro en los registros seleccionados, siempre muestra el nombre distinguido completo en un formato que se pueden reproducir más adelante.

```
dmpmqcfg -m QM -x authrec
------------------------------------
SET AUTHREC PROFILE(SELF) +
    PRINCIPAL('cn=JohnDoe, ou=users, o=yourcompany, c=yourcountry') +
    OBJTYPE(QMGR)
    AUTHADD(CONNECT)
```

Otras consideraciones al utilizar la autorización LDAP

Breve descripción de los cambios en la interfaz de cola de mensajes (Message Queue Interface, MQI) y otros mandatos MQSC y PCF necesarios para saber cuándo se usa una autorización LDAP desde IBM MQ 9.0.0.

ADOPTCTX

No hay ningún requisito para que las aplicaciones proporcionen información de autenticación, o para que el atributo ADOPTCTX se establezca en YES.

Si una aplicación no autentica explícitamente, o si **ADOPTCTX** se establece en NO para el objeto CONNAUTH activo, el contexto de identidad asociado a la aplicación se obtiene del ID de usuario del sistema operativo.

Cuando es necesario aplicar autorizaciones, dicho contexto se correlaciona con una entidad LDAP utilizando las mismas reglas que para los mandatos setmqaut.

Parámetros de entrada para llamadas MQI

MQOPEN, MQPUT1 y MQSUB tienen estructuras que permiten especificar un ID de usuario alternativo.

Si estos campos se utilizan, el ID de usuario de 12 caracteres se correlaciona con un nombre distinguido utilizando las mismas reglas que en los mandatos **setmqaut**, **dmpmqaut** y **dspmqaut**.

MQPUT y MQPUT1 también permiten programas con la autorización adecuada establecer el campo de MQMD UserIdentifier. El valor de este campo no se supervisará durante el proceso PUT y puede establecerse en cualquier valor.

Sin embargo, como es habitual, el valor **UserIdentifier** se puede utilizar para la autorización en fases posteriores del proceso de mensajes, por ejemplo cuando PUTAUT(CTX) se define en un canal receptor.

En dicho momento, se comprobará la autorización del identificador del receptor utilizando la configuración de dicho gestor de colas receptor, que puede ser basado en sistema operativo o LDAP.

Parámetros de salida para llamadas MQI

Siempre que se proporciona un ID de usuario a un programa en una estructura MQI, es la versión de nombre abreviado de 12 caracteres asociada a la conexión.

Por ejemplo, el valor **MQAXC.UserId** para las salidas de API es el nombre abreviado devuelto de la correlación LDAP.

Otros mandatos MQSC y PCF administrativos

Los mandatos que muestran información de usuario en el estado de objeto tales como, por ejemplo, DISPLAY CONN USERID devuelven el nombre abreviado de 12 caracteres asociado con el contexto. El nombre distinguido completo no se muestra.

Los mandatos que permiten la aserción de identidades, como las reglas de correlación CHLAUTH o los valores MCAUSER para canales, pueden tomar valores hasta llegar a la longitud máxima definida para estos atributos (actualmente 64 caracteres).

No hay ningún cambio en la sintaxis. Cuando es necesaria la autorización para esa identidad, ésta se correlaciona internamente con un nombre distinguido utilizando las mismas reglas que para los mandatos **setmqaut**, **dmpmqaut** y **dspmqaut**.

Esto significa que el valor MCAUSER en una definición de canal puede no visualizarse como la misma serie que DISPLAY CHSTATUS aunque hacen referencia a la misma identidad.

Por ejemplo:

```
DEFINE CHL(SV1) CHLTYPE(SVRCONN) MCAUSER('cn=JohnDoe')
```

```
DEFINE CHL(SV2) CHLTYPE(SVRCONN) MCAUSER('jodoe')
DEFINE CHL(SV3) CHLTYPE(SVRCONN) MCAUSER('JohnDoe1@yourcompany.com')
```

A continuación DISPLAY CHSTATUS(*) ALL mostrará el valor SHORTUSR, *MCAUSER(jodoe)* para todas las conexiones.

Conmutación entre modelos de autorización del sistema operativo y LDAP

Cómo se puede conmutar entre distintos métodos de autorización en distintas plataformas.

El atributo CONNAUTH del gestor de colas apunta a un objeto AUTHINFO. Cuando el objeto es del tipo IDPWLDAP, se utiliza un repositorio LDAP para la autenticación.

Ahora puede aplicar un método de autorización al mismo objeto, lo que le permite continuar con la autorización basada en sistema operativo, o trabajar con autorización LDAP

Plataformas UNIX e IBM i

El gestor de colas puede cambiarse en cualquier momento entre los modelos de sistema operativo y LDAP. Puede cambiar la configuración y hacer que dicha configuración sea la activa mediante el mandato REFRESH SECURITY TYPE (CONNAUTH).

Por ejemplo, si este objeto ya se ha configurado con la información de conexión para la autenticación:

```
ALTER AUTHINFO(MYLDAP) AUTHTYPE(IDPWLDAP) +
        AUTHORMD(SEARCHGRP) +
        BASEDNG('ou=groups,o=ibm,c=uk') +
        <other attributes>
ALTER QMGR CONNAUTH(MYLDAP)
REFRESH SECURITY
```

Windows

Si un cambio de configuración de autorización implica la conmutación entre modelos de sistema operativo y LDAP, el gestor de colas debe reiniciarse para que el cambio entre en vigor. de lo contrario, puede activar el cambio mediante el mandato REFRESH SECURITY TYPE (CONNAUTH).

Reglas de proceso

Cuando se cambia de la autorización de sistema operativo a la autorización LDAP, todas las reglas de autorización de sistema operativo existentes que se han establecido pasan a estar inactivas e invisibles.

Los mandatos tales como **dmpmqaut** no visualizan estas reglas de sistema operativo. De forma similar, cuando se vuelve a pasar de LDAP a SO, todas las autorizaciones LDAP definidas pasan a estar inactivas e invisibles, y se restauran las reglas de sistema operativo originales.

Si desea hacer copia de seguridad de las definiciones de un gestor de colas por alguna razón, con el mandato **dmpmqcfg**, esta copia de seguridad solamente incluirá las reglas definidas para el método de autorización en efecto en el momento de realizar la copia de seguridad.

Administración LDAP

Una visión general de cómo cada plataforma administra LDAP.

Cuando se utiliza la autorización LDAP, el miembro del grupo `mqm` (o equivalente) en el sistema operativo no es muy importante. Ser miembro de dicho grupo solamente controla si se pueden procesar determinados mandatos de línea de mandatos.

En concreto, debe ser miembro de dicho grupo para emitir mandatos strmqm y endmqm.

Una vez que el gestor de colas está en ejecución, ahora hay límites en la cuenta con todos los privilegios. Aparte del ID de usuario de la persona que emite el mandato **strmqm**, los otros usuarios que pertenecen al grupo mqm (o equivalente) del sistema operativo no tienen privilegios especiales.

Las autorizaciones de otros usuarios se basan en a qué grupos LDAP pertenecen. Un uso no calificado del nombre de grupo mqm en mandatos tales como **setmqaut** no está permitido para correlacionar con ningún grupo LDAP.

Plataformas UNIX

Una vez que el gestor de colas está en ejecución, la única cuenta que tiene todos los privilegios automáticamente es el usuario real que ha iniciado el gestor de colas.

El ID mqm sigue existiendo y se utiliza como el propietario de recursos del sistema operativo, tales como archivos, porque mqm es el ID efectivo bajo el que se ejecuta el gestor de colas. Sin embargo, el usuario mqm no podrá realizar automáticamente tareas administrativas controladas por el OAM.

IBM i

En IBM i, las cuentas que tienen automáticamente privilegios son las que inician el gestor de colas y el ID de QMQM.

Son necesarios los dos ID porque el ID de usuario que inicia el gestor de colas solamente es necesario para iniciar el sistema. Una vez que está en ejecución, los procesos del gestor de colas solo tienen autorización QMQM.

Plataformas Windows

En Windows, las cuentas con privilegios completos automáticos son la del usuario del sistema operativo que ha iniciado el gestor de colas, y también la del usuario que ejecuta los procesos principales del gestor de colas, como por ejemplo MUSR_MQADMIN si el gestor de colas se ha iniciado como un servicio de Windows.

Cuando se ejecuta en modalidad de autorización LDAP, Windows se comporta de forma muy parecida a las plataformas UNIX. Se ocupa de nombres abreviados de 12 caracteres y nombres distinguidos completos.

Script de ejemplo:

Como es útil tener un grupo que sea capaz de realizar la administración completa en un gestor de colas, se entrega un script de ejemplo en las plataformas UNIX como:

```
MQ_INSTALLATION_PATH/samp/bin/amqauthg.sh
```

Este ejemplo tiene dos parámetros:

- Un nombre de gestor de colas
- Un nombre de grupo LDAP

El ejemplo procesa mandatos setmqaut, y otorga autorización total para todos los objetos. Se trata del mismo script generado por el asistente de OAM de IBM MQ Explorer para funciones administrativas. Por ejemplo, el código empieza:

```
setmqaut -t q -m qmgr -n "**" +alladm +allmqi -g
      groupname
```

Confidencialidad de mensajes

Para mantener la confidencialidad, cifre los mensajes. Existen diversos métodos de cifrado de mensajes en IBM MQ, en función de sus necesidades.

Su elección de CipherSpec determina qué nivel de confidencialidad tiene.

Si necesita protección de datos de extremo a extremo a nivel de aplicación para la infraestructura de mensajería de punto a punto, puede utilizar Advanced Message Security para cifrar los mensajes o escribir su propia salida de API o salida cruzada de API.

Si necesita cifrar mensajes sólo mientras se están transportando por un canal, porque tiene seguridad adecuada en el gestor de colas, puede utilizar TLS, o puede escribir su propia salida de seguridad, salida de mensaje o programas de salida de emisión y recepción.

z/OS V 9.1.4 Si necesita cifrar los mensajes en reposo en un gestor de colas, puede utilizar el cifrado de conjunto de datos de z/OS en ese gestor de colas.

Para obtener más información sobre Advanced Message Security, consulte "Planificación de Advanced Message Security" en la página 105. El uso de TLS con IBM MQ se describe en "Protocolos de seguridad TLS en IBM MQ" en la página 24. El uso de programas de salida en el cifrado de mensaje se describe en "Implementación de confidencialidad en programas de salida de usuario" en la página 454.

Consulte la sección Confidencialidad para los datos en reposo en IBM MQ for z/OS con cifrado de conjunto de datos. para obtener más información sobre el cifrado de conjunto de datos de z/OS.

Tareas relacionadas

Conexión de dos gestores de colas utilizando TLS

Conexión de un cliente a un gestor de colas de forma segura

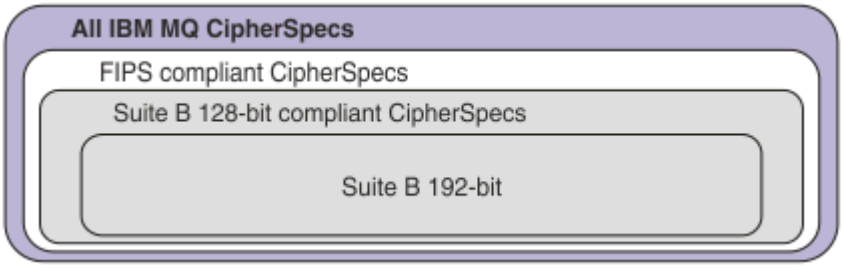
Habilitación de CipherSpecs

Habilite una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC **DEFINE CHANNEL** o el mandato MQSC **ALTER CHANNEL**.

Algunas de las CipherSpecs que puede utilizar con IBM MQ son compatibles con FIPS. Algunas de las CipherSpecs compatibles con FIPS también son compatibles con Suite B aunque otras, como TLS_RSA_WITH_AES_256_CBC_SHA, no lo son.

Todas las CipherSpecs compatibles con Suite B también son compatibles con FIPS.
Todas las CipherSpecs compatibles con Suite B se clasifican en dos grupos:
128 bits (por ejemplo, ECDHE_ECDSA_AES_128_GCM_SHA256 y 192 bits (por ejemplo, ECDHE_ECDSA_AES_256_GCM_SHA384),

El siguiente diagrama ilustra la relación entre estos subconjuntos:



A partir de IBM MQ 8.0.0 Fix Pack 3 , el número de CipherSpecs soportadas se ha reducido.

V 9.1.1 Si desea información sobre cómo configurar las CipherSpecs predeterminadas, consulte "Valores CipherSpec predeterminados habilitados en IBM MQ" en la página 430. También puede proporcionar un conjunto alternativo de CipherSpecs que están habilitadas para ser utilizadas con canales MQ. Consulte "Suministro de una lista personalizada de CipherSpecs habilitadas en multiplataformas" en la página 431.

Si desea más información sobre cómo habilitar las CipherSpecs en desuso, consulte "Habilitación de CipherSpecs en desuso en Multiplatforms" en la página 432 o "Habilitación de CipherSpecs en desuso en z/OS" en la página 432. Para obtener una lista de CipherSpecs que puede volver a habilitar para utilizarlas con IBM MQ, consulte "CipherSpecs en desuso" en la página 435.

ULW V 9.1.4 A partir de IBM MQ 9.1.4, IBM MQ da soporte al protocolo de seguridad TLS 1.3 en UNIX, Linux, and Windows. Para obtener información sobre la utilización de estas CipherSpecs, consulte "Utilización de TLS 1.3 en IBM MQ" en la página 430 y "IBM MQ MQI client y TLS 1.3" en la página 430.

CipherSpecs que se pueden utilizar con el soporte TLS de IBM MQ

Las especificaciones de cifrado que puede utilizar con al gestor de colas de IBM MQ se listan automáticamente en la siguiente tabla. Cuando solicite un certificado personal, especifique un tamaño de clave para el par de claves pública y privada. El tamaño de clave que se utiliza durante el reconocimiento TLS es el tamaño almacenado en el certificado a menos que esté determinado por la CipherSpec, tal como está indicado en la tabla.

Tabla 74. CipherSpecs que puede utilizar con el soporte TLS de IBM MQ

Soporte de plataforma "1" en la página 429	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 429	Suite B
V 9.1.4 V 9.1.4 **CipherSpecs de alias**							
Todo	ANY_TLS13_OR_HIGHER "3" en la página 429 "4" en la página 429 "5" en la página 429	No disponible	Negociado	Negociado	Negociado	Negociado	Negociado
Todo	ANY_TLS13 "4" en la página 429 "5" en la página 429 "6" en la página 429	No disponible	TLS 1.3	Negociado	Negociado	Negociado	Negociado
Todo	ANY_TLS12_OR_HIGHER "4" en la página 429 "5" en la página 429 "7" en la página 429	No disponible	Negociado	Negociado	Negociado	Negociado	Negociado
Todo	ANY_TLS12 "8" en la página 429	No disponible	TLS 1.2	Negociado	Negociado	Negociado	Negociado
Todo	ANY "9" en la página 429	No disponible	Negociado	Negociado	Negociado	Negociado	Negociado
V 9.1.4 V 9.1.4 **CipherSpecs for TLS 1.3**							
Todo	TLS_AES_128_GCM_SHA256 "4" en la página 429	1301	TLS 1.3	GCM	AES-128 con GCM (128)	Sí	No
Todo	TLS_AES_256_GCM_SHA384 "4" en la página 429	1302	TLS 1.3	GCM	AES-256 con GCM (256)	Sí	No

Tabla 74. CipherSpecs que puede utilizar con el soporte TLS de IBM MQ (continuación)

Soporte de plataforma "1" en la página 429	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 429	Suite B
Todo	TLS_CHACHA20_POLY1305_SHA256 "4" en la página 429	1303	TLS 1.3	POLY1305	CHACHA20 (256)	No	No
ULW	TLS_AES_128_CCM_SHA256	1304	TLS 1.3	CBC-MAC	AES-128 con CTR (128)	Sí	No
ULW	TLS_AES_128_CCM_8_SHA256 "11" en la página 429	1305	TLS 1.3	CBC-MAC	AES-128 con CTR (128)	Sí	No
CipherSpecs para TLS 1.2							
Todo	TLS_RSA_WITH_AES_128_CBC_SHA256 "10" en la página 429	003C	TLS 1.2	SHA-256	AES (128)	Sí	No
Todo	TLS_RSA_WITH_AES_256_CBC_SHA256 "10" en la página 429 "12" en la página 429	003D	TLS 1.2	SHA-256	AES (256)	Sí	No
Todo	TLS_RSA_WITH_AES_128_GCM_SHA256 "10" en la página 429 "13" en la página 429	009C	TLS 1.2	SHA-256 y AEAD GCM	AES (128)	Sí	No
Todo	TLS_RSA_WITH_AES_256_GCM_SHA384 "10" en la página 429 "12" en la página 429 "13" en la página 429	009D	TLS 1.2	SHA-384 y AEAD GCM	AES (256)	Sí	No
Todo	ECDHE_ECDSA_AES_128_CBC_SHA256 "10" en la página 429	C023	TLS 1.2	SHA-256	AES (128)	Sí	No
Todo	ECDHE_ECDSA_AES_256_CBC_SHA384 "10" en la página 429 "12" en la página 429	C024	TLS 1.2	SHA-384	AES (256)	Sí	No
Todo	ECDHE_RSA_AES_128_CBC_SHA256 "10" en la página 429	C027	TLS 1.2	SHA-256	AES (128)	Sí	No
Todo	ECDHE_RSA_AES_256_CBC_SHA384 "10" en la página 429 "12" en la página 429	C028	TLS 1.2	SHA-384	AES (256)	Sí	No
Multi	ECDHE_ECDSA_AES_128_GCM_SHA256 "12" en la página 429 "13" en la página 429	C02B	TLS 1.2	SHA-256 y AEAD GCM	AES (SHA384)	Sí	128 bits
Multi	ECDHE_ECDSA_AES_256_GCM_SHA384 "12" en la página 429 "13" en la página 429	C02C	TLS 1.2	SHA-384 y AEAD GCM	AES (SHA384)	Sí	192 bits
Todo	ECDHE_RSA_AES_128_GCM_SHA256 "13" en la página 429	C02F	TLS 1.2	SHA-256 y AEAD GCM	AES (128)	Sí	No
Todo	ECDHE_RSA_AES_256_GCM_SHA384 "12" en la página 429 "13" en la página 429	C030	TLS 1.2	AEAD AES-128 GCM	AES (SHA384)	Sí	No

Tabla 74. CipherSpecs que puede utilizar con el soporte TLS de IBM MQ (continuación)

Soporte de plataforma "1" en la página 429	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 429	Suite B

Notas:

1. Para obtener una lista de plataformas cubiertas por cada icono de plataforma, consulte Iconos de release y plataforma en la documentación del producto.
2. Especifica si la CipherSpec tiene el certificado FIPS en una plataforma certificada con FIPS. Consulte Federal Information Processing Standards (FIPS - Estándares federales de procesamiento de información) para obtener una explicación de FIPS.
3. ULW El alias ANY_TLS13_OR_HIGHER de CipherSpec negocia el mayor nivel de seguridad que el extremo remoto permitirá, pero solo se conectará utilizando un protocolo TLS 1.3 o superior.
4. z/OS Para utilizar TLS 1.3, o el CipherSpec ANY, en IBM MQ for z/OS, el sistema operativo debe ser z/OS 2.4 o posterior.
5. IBM i Para utilizar TLS 1.3 en IBM i, la versión del sistema operativo subyacente debe soportar TLS 1.3. Consulte Soporte del sistema TLS para TLSv1.3 para obtener más información.
6. ULW El alias ANY_TLS13 de CipherSpec representa un subconjunto de CipherSpecs aceptables que utilizan el protocolo TLS 1.3, como se enumeran en esta tabla para cada plataforma.
7. ULW El alias ANY_TLS12_OR_HIGHER de CipherSpec negocia el mayor nivel de seguridad que el extremo remoto permitirá, pero solo se conectará utilizando un protocolo TLS 1.2 o superior.
8. La ANY_TLS12 CipherSpec representa un subconjunto de CipherSpecs aceptables que utilizan el protocolo TLS 1.2, como aparecen listadas en esta tabla para cada plataforma.
9. ULW El alias ANY de CipherSpec negocia el mayor nivel de seguridad que el extremo remoto permitirá.
10. IBM i Estas CipherSpecs no están habilitadas en los sistemas IBM i 7.4 que tienen el valor del sistema QSSLCSLCTL establecido en *OPSSYS.
11. ULW Estas CipherSpecs utilizan un valor de comprobación de integridad (ICV) de 8 octetos en lugar de un ICV de 16 octetos.
12. Esta CipherSpec no se puede utilizar para garantizar una conexión desde IBM MQ Explorer a un gestor de colas amenos que se apliquen los archivos de políticas no restringidas apropiados al JRE utilizado por Explorer.
13. Windows Linux Siguiendo una recomendación de GSKit, TLS 1.2 GCM CipherSpecs tienen una restricción que significa que después de que se envíen los registros TLS24.5 , utilizando la misma clave de sesión, la conexión se termina con el mensaje AMQ9288E. Esta restricción de GCM está activa, independientemente de la modalidad FIPS que se esté utilizando.

    Para evitar que se produzca este error, evite utilizar cifrados TLS 1.2 GCM , habilite el restablecimiento de la clave secreta o inicie el cliente o el gestor de colas de IBM MQ con la variable de entorno `GSK_ENFORCE_GCM_RESTRICTION=GSK_FALSE` establecida. Para las bibliotecas de GSKit , debe establecer esta variable de entorno en ambos lados de la conexión y aplicarla a las conexiones de cliente a gestor de colas y al gestor de colas a las conexiones de gestor de colas. Tenga en cuenta que este valor afecta a los clientes .NET no gestionados, pero no a los clientes Java o gestionados .NET . Para obtener más información, consulte AES-GCM restricción de cifrado.

    Esta restricción no es aplicable a IBM MQ for z/OS.

Utilización de TLS 1.3 en IBM MQ

ULW V 9.1.4

A partir de IBM MQ 9.1.4, IBM MQ da soporte a TLS 1.3 en UNIX, Linux, and Windows. En cualquier instalación soportada, los gestores de colas nuevos se crean con una entrada en la stanza SSL del archivo `qm.ini` que lee:

```
SSL:
  AllowTLSV13=TRUE
```

Nota: El archivo `qm.ini` se puede encontrar en el directorio *<data directory>*`/qmgrs/`*<qmgr name>*.

Si el gestor de colas se ha creado utilizando una versión de IBM MQ anterior a IBM MQ 9.1.4, pero más adelante se inicia utilizando IBM MQ 9.1.4 o superior, no tendrá el conjunto de propiedades **`AllowTLSV13`**. Si desea habilitar TLS 1.3, debe editar `qm.ini file` y añadir la propiedad tal como se muestra en el ejemplo (incluida la stanza "SSL:" si todavía no existe).

Esta propiedad de archivo .ini habilita TLS 1.3, que permite el uso de CipherSpecs de TLS 1.3. De acuerdo con la especificación de TLS 1.3, cualquier intento de comunicarse con una CipherSpec débil, independientemente de si están o no habilitadas en IBM MQ, se rechazará. Las CipherSpecs que TLS 1.3 considera que son débiles son CipherSpecs que cumplen uno o varios de los criterios siguientes:

- Utiliza el protocolo SSL 3.0.
- Utiliza RC4 o RC2 como algoritmo de cifrado.
- Tiene un tamaño de clave de cifrado (bit) igual o inferior a 112.

Estas restricciones se indican con la nota [10] en la Tabla 1 de las CipherSpecs en desuso.

Si debe continuar utilizando estas CipherSpecs, debe inhabilitar la modalidad TLS 1.3. Esto se hace editando el archivo `qm.ini` del gestor de colas y cambiando el valor de la propiedad **`AllowTLSV13`** a:

```
SSL:
  AllowTLSV13=FALSE
```

Nota: Cuando se aplique este valor, utilice CipherSpecs de TLS 1.3.

IBM MQ MQI client y TLS 1.3

ULW V 9.1.4

Cuando se utiliza el cliente de IBM MQ MQI client, el valor de **`AllowTLSV13`** se infiere, a menos que se especifique explícitamente en la stanza SSL del archivo `mqclient.ini` que está utilizando la aplicación.

- Si las CipherSpecs débiles están habilitadas, **`AllowTLSV13`** se establece en FALSE y no se puede utilizar ninguna CipherSpecs de TLS 1.3.
- De lo contrario, **`AllowTLSV13`** se establece en TRUE y se pueden utilizar las nuevas CipherSpecs y CipherSpecs de alias de TLS 1.3.

Valores CipherSpec predeterminados habilitados en IBM MQ

V 9.1.1 Multi

En la configuración predeterminada, IBM MQ proporciona soporte para el protocolo TLS 1.2 y distintos algoritmos criptográficos que utilizan CipherSpecs. A efectos de compatibilidad, IBM MQ también se puede configurar para utilizar protocolos SSL 3.0 y TLS 1.0 y una serie de algoritmos de cifrado que se sabe que son débiles o susceptibles de vulnerabilidades de seguridad. La lista de CipherSpecs que están habilitadas en la configuración predeterminada puede cambiar aplicando el mantenimiento.

Es posible configurar IBM MQ para restringir o permitir el uso de CipherSpecs utilizando los controles siguientes:

- Solo permite CipherSpecs compatibles con FIPS 140-2 utilizando SSLFIPS.

- ULW Solo permitir CipherSpecs compatibles con NSA Suite B utilizando SUITEB.
- ULW Permitir una lista personalizada de CipherSpecs utilizando **AllowedCipherSpecs** o la variable de entorno **AMQ_ALLOWED_CIPHERS** .
- ULW Permitir el uso de CipherSpecs en desuso utilizando **AllowWeakCipher** o la variable de entorno **AMQ_SSL_WEAK_CIPHER_ENABLE**.
- z/OS Permitir el uso de las CipherSpecs en desuso utilizando sentencias DD en el JCL CHINIT.

Nota: Si especifica una lista personalizada de CipherSpecs utilizando **AllowedCipherSpecs** o **AMQ_ALLOWED_CIPHERS**, esto altera temporalmente la habilitación de cualquier CipherSpecs en desuso. Tenga en cuenta que al utilizar las restricciones de NSA Suite B o FIPS 140-2 en combinación con una lista de CipherSpec personalizada, debe asegurarse de que la lista personalizada solo contiene las CipherSpecs permitidas por los valores de Suite B o FIPS 140-2.

Suministro de una lista personalizada de CipherSpecs habilitadas en multiplataformas

V 9.1.1 Multi

Es posible que proporcione un conjunto alternativo de CipherSpecs que estén habilitadas para su uso con canales IBM MQ, ya sea utilizando la variable de entorno **AMQ_ALLOWED_CIPHERS** o el atributo **AllowedCipherSpecs** de stanza SSL del archivo `.ini`. Es posible que desee utilizar este valor para restringir a los escuchas de IBM MQ la aceptación de solicitudes de inicio de canal de entrada, a menos que utilicen una de las CipherSpecs especificadas. Se puede utilizar esta funcionalidad para controlar las CipherSpecs incluidas en las CipherSpecs ANY*.

La variable de entorno **AMQ_ALLOWED_CIPHERS** o el atributo **AllowedCipherSpecs** de stanza SSL acepta:

- Un solo nombre de CipherSpec o
- Una lista separada por comas de nombres de CipherSpec de IBM MQ para volver a habilitar, o
- El valor especial de ALL, que representa todas las CipherSpecs (no se recomienda).

Nota: No se recomienda habilitar **ALL** (todas las) CipherSpecs ya que esto habilitará los protocolos SSL 3.0 y TLS 1.0 y un gran número de algoritmos criptográficos débiles.

Si se ha configurado este valor, sustituye la lista de CipherSpec predeterminada y hace que IBM MQ ignore los valores de cifrado débiles en desuso (consulte más abajo):

- Los escuchas de IBM MQ solo aceptarán propuestas SSL/TLS que utilicen una de las CipherSpecs especificadas.
- Los canales de IBM MQ sólo permitirán un valor SSLCIPH en blanco, o una de las CipherSpecs especificadas.
- La finalización del separador **runmqsc** de los valores SSLCIPH restringe los valores de terminación a una de las CipherSpecs especificadas.

Por ejemplo, si solo desea permitir que los canales se definan/alteren y que los escuchas acepten ECDHE_RSA_AES_128_GCM_SHA256 o ECDHE_ECDSA_AES_256_GCM_SHA384, podría establecer lo siguiente en el archivo qm.ini:

```
SSL:
  AllowedCipherSpecs=ECDHE_RSA_AES_128_GCM_SHA256, ECDHE_ECDSA_AES_256_GCM_SHA384
```

Tenga en cuenta que los cifrados utilizados por canales AMQP o MQTT se pueden restringir utilizando los valores del archivo java.security.

Habilitación de CipherSpecs en desuso en Multiplatforms

Multi

De forma predeterminada, no está permitido especificar una CipherSpec en desuso en una definición de canal. Si intenta especificar una CipherSpec en desuso en Multiplatforms, recibirá el mensaje AMQ8242: definición SSLCIPH errónea y PCF devuelve MQRCCF_SSL_CIPHER_SPEC_ERROR.

No puede iniciar un canal con una CipherSpec en desuso. Si intenta hacerlo con una CipherSpecen desuso, el sistema devuelve MQCC_FAILED (2), junto con un **Reason** de MQRC_SSL_INITIALIZATION_ERROR (2393) al cliente.

Puede volver a habilitar una o más de las CipherSpecs en desuso para definir canales, en tiempo de ejecución en el servidor, estableciendo la variable de entorno **AMQ_SSL_WEAK_CIPHER_ENABLE**.

La variable de entorno **AMQ_SSL_WEAK_CIPHER_ENABLE** acepta:

- Un solo nombre de CipherSpec o
- Una lista separada por comas de nombres de CipherSpec de IBM MQ para volver a habilitar, o
- El valor especial de ALL, que representa todas las CipherSpecs (no se recomienda).

Nota: No se recomienda volver a habilitar ALL CipherSpecs, ya que esto habilitará los protocolos SSL 3.0 y TLS 1.0 y un gran número de algoritmos de cifrado débiles.

Por ejemplo, si desea volver a habilitar ECDHE_RSA_RC4_128_SHA256, establezca la siguiente variable de entorno:

```
export AMQ_SSL_WEAK_CIPHER_ENABLE=ECDHE_RSA_RC4_128_SHA256
```

o, como alternativa, cambie la stanza SSL en el archivo qm.ini estableciendo:

```
SSL:
  AllowTLSV1=Y
  AllowWeakCipherSpec=ECDHE_RSA_RC4_128_SHA256
```

Habilitación de CipherSpecs en desuso en z/OS

z/OS

De forma predeterminada, no está permitido especificar una CipherSpec en desuso en una definición de canal. Si intenta especificar una CipherSpec en desuso en z/OS, recibirá un mensaje CSQM102E o un mensaje CSQX674E.

Si habilita cipherspecs débiles (en desuso), deberá definir la siguiente sentencia DD en el JCL de CHINIT:

```
JCL: //CSQXWEAK DD DUMMY
```

Nota: No todas las CipherSpecs en desuso requieren el uso de esta sentencia DD, consulte la nota 11 de la tabla dentro de “CipherSpecs en desuso” en la página 435.

Para habilitar el protocolo SSL 3.0 en desuso, también es necesario que defina la siguiente sentencia DD en el JCL de CHINIT:

```
JCL: //CSQXSSL3 DD DUMMY
```

V 9.1.0 Para habilitar el protocolo TLS 1.0 en desuso, también es necesario que defina la siguiente sentencia DD en el JCL de CHINIT:

```
JCL: //TLS10ON DD DUMMY
```

Tenga en cuenta que el nombre de la tarjeta DD es TLS10ON, para indicar que TLS 1.0 está activado (ON) y no TLS100N.

Para desactivar TLS 1.0 (OFF), utilice la sentencia siguiente:

```
JCL: //TLS10OFF DD DUMMY
```

Si no desea negociar con el escucha utilizando especificaciones de cifrado débiles o rotas, debe definir la siguiente sentencia DD en el JCL de CHINIT:

```
JCL: //WCIPSOFF DD DUMMY
```

Si sólo desea negociar con el escucha utilizando las especificaciones de cifrado listadas en la lista de especificaciones de cifrado predeterminadas de **System SSL** , debe definir la siguiente sentencia DD en el JCL de CHINIT:

```
JCL: //GSKDCIPS DD DUMMY
```

Nivel mínimo frente a CipherSpecs de nivel fijo

ULW V 9.1.4

IBM MQ da soporte a dos tipos distintos de CipherSpecs:

- Las CipherSpecs de **nivel mínimo** son las que no establecen un límite superior, por ejemplo ANY, ANY_TLS12_OR_HIGHER o ANY_TLS13_OR_HIGHER.
- Las CipherSpecs de **nivel fijo** son las que identifican un protocolo específico, por ejemplo ANY_TLS12 y ANY_TLS13, o un algoritmo específico como, por ejemplo, ECDHE_ECDSA_3DES_EDE_CBC_SHA256

Para maximizar la simplicidad de la configuración al tiempo que se mantiene la seguridad, se recomienda el uso de CipherSpecs de **nivel mínimo** en ambos lados del canal. Esto permite que las comunicaciones puedan dar soporte automáticamente a una versión de protocolo TLS superior, y utilizarlo, cuando ambos lados dan soporte a una nueva versión sin necesidad de cambiar la configuración de ambos lados.

El uso de un **nivel mínimo** CipherSpec en el lado iniciador, pero un **nivel fijo** CipherSpec en el lado receptor podría hacer que la conexión se rechazara y que se emitieran los mensajes AMQ9631 y AMQ9641 .

Consulte “Relación entre los valores CipherSpec de alias” en la página 439 para ver las tablas que contienen distintos resultados para los valores de Alias CipherSpec .

Conceptos relacionados

“Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45
En este tema se proporciona información sobre cómo elegir las CipherSpecs y los certificados digitales adecuados para su política de seguridad, describiendo la relación entre las CipherSpecs y los certificados digitales en IBM MQ.

“CipherSpecs y CipherSuites” en la página 19
Los protocolos de seguridad criptográficos deben estar de acuerdo con los algoritmos utilizados por una conexión segura. CipherSpecs y CipherSuites definen combinaciones específicas de algoritmos.

“Configuración de IBM MQ para Suite B” en la página 43
IBM MQ se puede configurar para que funcione en conformidad con el estándar Suite B de la NSA en las plataformas Windows, UNIX and Linux.

“Estándares federales de procesamiento de la información (FIPS)” en la página 33
En este tema se presenta los estándares federales de procesamiento de la información (FIPS) Cryptomodule Validation Program del US National Institute of Standards and Technology y las funciones de cifrado que se pueden utilizar en canales TLS.

Tareas relacionadas

Migración de configuraciones de seguridad existentes para utilizar la CipherSpec ANY_TLS12_OR_HIGHER

Referencia relacionada

DEFINE CHANNEL

ALTER CHANNEL
Cambiar, Copiar y Crear canal

ULW AES-Restricción de cifrado deGCM

Una guía de las restricciones que se imponen en los cifrados AES-GCM cuando se utilizan para criptografía TLS. Estas restricciones las imponen las organizaciones IETF y NIST y requieren que no se utilice la misma clave de sesión para transferir de forma segura más de 2 registros[24.5] TLS cuando se utilizan cifrados AES-GCM .

Para obtener más información sobre estas restricciones, consulte Sección RFC 9325 4.4 Limits on Key Usage y Sección RFC 8446 5.5.

IBM MQ no implementa la funcionalidad criptográfica directamente. En su lugar, se utilizan varias bibliotecas criptográficas diferentes para proporcionar la funcionalidad TLS y Advanced Message Security . En los sistemas operativos Windows, Linuxy AIX , la biblioteca criptográfica que IBM MQ utiliza es GSKit. Para las aplicaciones, las bibliotecas C y .NET no gestionadas utilizan GSKit para la funcionalidad criptográfica. La implementación de los algoritmos de cifrado AES-GCM mediante GSKit incluye las restricciones especificadas por el grupo de estándares. Además, estas restricciones están habilitadas de forma predeterminada. Como tal, la comunicación TLS de IBM MQ , al utilizar cifrados AES-GCM , termina si se transmiten más de 2 registros TLS[24.5] utilizando la misma clave de sesión.

Nota: Esta restricción no está presente en las plataformas IBM i, IBM Z o IBM MQ for HPE NonStop o Java/JMS, las aplicaciones .NET gestionadas porque se utilizan distintas bibliotecas criptográficas y estas bibliotecas no han implementado la misma restricción.

Si un canal IBM MQ permanece en ejecución durante el tiempo suficiente para que se transmitan más de 2 registros TLS[24.5] utilizando la misma clave de sesión, la biblioteca criptográfica subyacente termina la conexión. Esto hace que el canal termine y se genere un mensaje de error AMQ9288E . Las aplicaciones que tienen su comunicación terminada de este modo reciben un código de retorno `MQRC_CONNECTION_BROKEN` de la operación IBM MQ que se estaba realizando.

La terminación de la conexión se puede realizar en cualquiera de los dos extremos de la comunicación, pero sólo en los extremos que utilizan GSKit para la funcionalidad criptográfica.

Consejos para mitigar la restricción

Algunas opciones sobre cómo evitar o manejar las comunicaciones que se terminan debido a esta restricción son las siguientes:

Utilizar clientes reconectables

Las aplicaciones se pueden configurar para que intenten automáticamente una reconexión, en caso de que falle una conexión. Esto incluye las conexiones que han terminado debido a la restricción GCM . Cuando se configura para la reconexión, la aplicación cliente se restaura automáticamente en cualquier punto de anomalía y se restaura cualquier descriptor de contexto para abrir objetos. Esto se hace sin volver al código de aplicación.

Para obtener más información, consulte Reconexión de cliente automática.

Establecer un valor de restablecimiento de clave secreta

IBM MQ se puede configurar para solicitar un restablecimiento de clave de sesión después de que se haya transferido un número configurable de bytes a través de un canal. Al alcanzar este límite, IBM MQ solicita que la capa criptográfica realice un restablecimiento de clave de sesión, lo que da como resultado una nueva clave de sesión.

Es importante tener en cuenta que el valor especificado es el número de bytes transferidos, que se relaciona con el tamaño de los mensajes enviados por IBM MQ. La restricción está en el número de registros TLS que se envían. No hay una correlación directa entre los bytes de mensajes y los registros TLS, ya que un registro TLS puede enviar un número máximo de bytes que depende de la unidad máxima de transmisión (MTU) de la red. Los mensajes que se envían que son mayores que este valor se transmiten como varios registros TLS. El valor de MTU varía entre redes. Además, existen otras razones por las que un registro TLS puede tener que enviarse fuera de la transmisión de datos de

mensajes de IBM MQ , por ejemplo, IBM MQ comprobaciones de latido, alertas TLS, otros mensajes de protocolo IBM MQ . Estos registros TLS adicionales cuentan para el número máximo de registros TLS, pero no se cuentan en el valor de restablecimiento de clave secreta de IBM MQ .

El restablecimiento regular de una clave de sesión utilizando el restablecimiento de clave secreta puede impedir que el canal termine debido a la restricción AES-GCM .

Para obtener más información, consulte Restablecimiento de claves secretas SSL y TLS.

V 9.1.4 **Utilizar especificaciones de cifrado TLS 1.3**

Aunque la restricción AES-GCM sigue estando presente cuando se utiliza el protocolo TLS 1.3 , el protocolo TLS 1.3 da soporte a la realización automática de un restablecimiento de clave de sesión sin necesidad de interrumpir las comunicaciones TLS. Esto permite a GSKit gestionar el restablecimiento de la clave de sesión cuando sea necesario sin que IBM MQ necesite solicitar un restablecimiento de clave secreta.

Para obtener más información, consulte Utilización de TLS 1.3 en IBM MQ en “Habilitación de CipherSpecs” en la página 426.

Inhabilitar la restricción AES-GCM

Si es necesario, la restricción se puede inhabilitar estableciendo la variable de entorno **GSK_ENFORCE_GCM_RESTRICTION=GSK_FALSE** para inhabilitar la restricción AES-GCM . Esto permite enviar cualquier número de registros TLS utilizando la misma clave de sesión. Si elige esta mitigación, la variable de entorno debe establecerse en cada extremo de la comunicación que utiliza GSKit para las comunicaciones seguras.

Aviso: Esta opción no se recomienda ya que, después de que se hayan enviado más de 2 registros TLS[24.5] , es posible que los atacantes realicen análisis en los registros enviados para determinar la clave de sesión en uso. Una vez que se ha determinado la clave de sesión, todas las comunicaciones existentes y futuras que utilizan dicha clave de sesión se ven comprometidas.

CipherSpecs en desuso

Una lista de CipherSpecs en desuso que puede utilizar con IBM MQ si es necesario.

Si desea más información sobre cómo habilitar las CipherSpecs en desuso, consulte “Habilitación de CipherSpecs en desuso en Multiplatforms” en la página 432 o “Habilitación de CipherSpecs en desuso en z/OS” en la página 432.

En la siguiente tabla se listan las CipherSpecs en desuso que puede utilizar con el soporte TLS de IBM MQ.

Tabla 75. CipherSpecs en desuso que puede volver a habilitar para su uso con IBM MQ

Soporte de plataforma “1” en la página 438	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS “2” en la página 438	Suite B	Actualizar cuando esté en desuso
CipherSpecs para SSL 3.0								
IBM i	AES_SHA_US “3” en la página 438	002F	SSL 3.0	SHA-1	AES (128)	No	No	9.0.0.0
Todo	DES_SHA_EXPORT “3” en la página 438 “4” en la página 438 “5” en la página 438	0009	SSL 3.0	SHA-1	DES (56)	No	No	9.0.0.0
ULW	DES_SHA_EXPORT1024 “3” en la página 438 “6” en la página 438	0062	SSL 3.0	SHA-1	DES (56)	No	No	9.0.0.0

Tabla 75. CipherSpecs en desuso que puede volver a habilitar para su uso con IBM MQ (continuación)

Soporte de plataforma "1" en la página 438	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 438	Suite B	Actualizar cuando esté en desuso
ULW	FIPS_WITH_DES_CBC_SHA "3" en la página 438	FEFE	SSL 3.0	SHA-1	DES (56)	No "7" en la página 438	No	9.0.0.0
ULW	FIPS_WITH_3DES_EDE_CBC_SHA "3" en la página 438	FEFF	SSL 3.0	SHA-1	3DES (168)	No "8" en la página 438	No	9.0.0.1 y 9.0.1
Todo	NULL_MD5 "3" en la página 438	0001	SSL 3.0	MD5	Ninguna	No	No	9.0.0.1
Todo	NULL_SHA "3" en la página 438	0002	SSL 3.0	SHA-1	Ninguna	No	No	9.0.0.1
Todo	RC2_MD5_EXPORT "3" en la página 438 "4" en la página 438 "5" en la página 438	0006	SSL 3.0	MD5	RC2 (40)	No	No	9.0.0.0
Todo	RC4_MD5_EXPORT "4" en la página 438 "3" en la página 438	0003	SSL 3.0	MD5	RC4 (40)	No	No	9.0.0.0
Todo	RC4_MD5_US "3" en la página 438	0004	SSL 3.0	MD5	RC4 (128)	No	No	9.0.0.0
Todo	RC4_SHA_US "3" en la página 438 "5" en la página 438	0005	SSL 3.0	SHA-1	RC4 (128)	No	No	9.0.0.0
ULW	RC4_56_SHA_EXPORT1024 "3" en la página 438 "6" en la página 438	0064	SSL 3.0	SHA-1	RC4 (56)	No	No	9.0.0.0
Todo	TRIPLE_DES_SHA_US "3" en la página 438 "5" en la página 438	000A	SSL 3.0	SHA-1	3DES (168)	No	No	9.0.0.1 y 9.0.1
CipherSpecs para TLS 1.0								
IBM i	TLS_RSA_EXPORT_WITH_RC2_40_MD5 "3" en la página 438	0006	TLS 1.0	MD5	RC2 (40)	No	No	9.0.0.0
IBM i	TLS_RSA_EXPORT_WITH_RC4_40_MD5 "3" en la página 438 "4" en la página 438	0003	TLS 1.0	MD5	RC4 (40)	No	No	9.0.0.0
Todo	TLS_RSA_WITH_DES_CBC_SHA "3" en la página 438	0009	TLS 1.0	SHA-1	DES (56)	No "9" en la página 438	No	9.0.0.0
IBM i	TLS_RSA_WITH_NULL_MD5 "3" en la página 438	0001	TLS 1.0	MD5	Ninguna	No	No	9.0.0.1
IBM i	TLS_RSA_WITH_NULL_SHA "3" en la página 438	0002	TLS 1.0	SHA-1	Ninguna	No	No	9.0.0.1
IBM i	TLS_RSA_WITH_RC4_128_MD5 "3" en la página 438	0004	TLS 1.0	MD5	RC4 (128)	No	No	9.0.0.0

Tabla 75. CipherSpecs en desuso que puede volver a habilitar para su uso con IBM MQ (continuación)

Soporte de plataforma "1" en la página 438	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 438	Suite B	Actualizar cuando esté en desuso
z/OS ULW	TLS_RSA_WITH_AES_128_CBC_SHA "10" en la página 438	002F	TLS 1.0	SHA-1	AES (128)	Sí	No	9.0.5
z/OS ULW	TLS_RSA_WITH_AES_256_CBC_SHA "6" en la página 438 "10" en la página 438	0035	TLS 1.0	SHA-1	AES (256)	Sí	No	9.0.5
Todo	TLS_RSA_WITH_3DES_EDE_CBC_SHA	000A	TLS 1.0	SHA-1	3DES (168)	Sí	No	9.0.0.1 y 9.0.1
CipherSpecs para TLS 1.2								
ULW	ECDHE_ECDSA_NULL_SHA256 "3" en la página 438	C006	TLS 1.2	SHA-1	Ninguna	No	No	9.0.0.1
ULW	ECDHE_ECDSA_RC4_128_SHA256 "3" en la página 438	C007	TLS 1.2	SHA-1	RC4 (128)	No	No	9.0.0.0
IBM i ULW	ECDHE_RSA_NULL_SHA256 "3" en la página 438	C010	TLS 1.2	SHA-1	Ninguna	No	No	9.0.0.1
IBM i ULW	ECDHE_RSA_RC4_128_SHA256 "3" en la página 438	C011	TLS 1.2	SHA-1	RC4 (128)	No	No	9.0.0.0
ULW	TLS_RSA_WITH_NULL_NULL "3" en la página 438	0000	TLS 1.2	Ninguna	Ninguna	No	No	9.0.0.1
Todo	TLS_RSA_WITH_NULL_SHA256 "3" en la página 438	003B	TLS 1.2	SHA-256	Ninguna	No	No	9.0.0.1
ULW	TLS_RSA_WITH_RC4_128_SHA256 "3" en la página 438	0005	TLS 1.2	SHA-1	RC4 (128)	No	No	9.0.0.0
ULW	ECDHE_ECDSA_3DES_EDE_CBC_SHA256	C0008	TLS 1.2	SHA-1	3DES (168)	Sí	No	9.0.0.1 y 9.0.1
IBM i ULW	ECDHE_RSA_3DES_EDE_CBC_SHA256	C012	TLS 1.2	SHA-1	3DES (168)	Sí	No	9.0.0.1 y 9.0.1

Tabla 75. CipherSpecs en desuso que puede volver a habilitar para su uso con IBM MQ (continuación)

Soporte de plataforma "1" en la página 438	Nombre de CipherSpec	Código hexadecimal	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado (bits de cifrado)	FIPS "2" en la página 438	Suite B	Actualizar cuando esté en desuso

Notas:

1. Para obtener una lista de plataformas cubiertas por cada icono de plataforma, consulte Iconos de release y plataforma en la documentación del producto.
2. Especifica si la CipherSpec tiene el certificado FIPS en una plataforma certificada con FIPS. Consulte Federal Information Processing Standards (FIPS - Estándares federales de procesamiento de información) para obtener una explicación de FIPS.
3. ULW Estas CipherSpecs están inhabilitadas cuando TLS 1.3 está habilitado (a través de la propiedad AllowTLSV13 en `qm.ini`).

   z/OS Los gestores de colas creados en IBM MQ for z/OS 9.2.0 o posterior habilitan TLS 1.3 de forma predeterminada, lo que inhabilita estas CipherSpecs. Puede habilitar estas CipherSpecs, si es necesario, desactivando TLS V1.3. Esto se lleva a cabo añadiendo **AllowTLSV13**=*FALSE* a la stanza TransportSecurity del conjunto de datos QMINI en el JCL del gestor de colas. Los gestores de colas migrados a IBM MQ for z/OS 9.2.0 desde una versión anterior no tienen TLS 1.3 habilitado de forma predeterminada y, por lo tanto, tienen estas CipherSpecs habilitadas.
4. El tamaño máximo de la clave de reconocimiento es de 512 bits. Si cualquiera de los certificados intercambiados durante el reconocimiento SSL tiene un tamaño de clave mayor de 512 bits, se genera una clave temporal de 512 bits para poder utilizarla durante el reconocimiento.
5. IBM MQ classes for Java o IBM MQ classes for JMS ya no soportan estas CipherSpecs. Para obtener más información, consulte CipherSpecs y CipherSuites SSL/TLS en IBM MQ classes for Java o CipherSpecs y CipherSuites SSL/TLS en IBM MQ classes for JMS.
6. El tamaño de clave de reconocimiento es de 1024 bits.
7. Esta CipherSpec obtuvo el certificado FIPS 140-2 antes del 19 mayo de 2007. El nombre `FIPS_WITH_DES_CBC_SHA` es histórico y refleja el hecho de que este CipherSpec era anteriormente (pero ya no lo es) compatible con FIPS. Esta CipherSpec está en desuso y su uso no se recomienda.
8. El nombre `FIPS_WITH_3DES_EDE_CBC_SHA` es histórico y refleja el hecho de que este CipherSpec era anteriormente (pero ya no lo es) compatible con FIPS. Esta CipherSpec está en desuso.
9. Esta CipherSpec obtuvo el certificado FIPS 140-2 antes del 19 mayo de 2007.
10. z/OS Volver a habilitar sólo estas CipherSpecs no requiere el uso de la sentencia CSQXWEAK DD.

Conceptos relacionados

"Certificados digitales y compatibilidad de CipherSpec en IBM MQ" en la página 45
En este tema se proporciona información sobre cómo elegir las CipherSpecs y los certificados digitales adecuados para su política de seguridad, describiendo la relación entre las CipherSpecs y los certificados digitales en IBM MQ.

Referencia relacionada

DEFINE CHANNEL
ALTER CHANNEL

Relación entre los valores CipherSpec de alias

Las tablas siguientes muestran el comportamiento esperado cuando TLS1.3 no está habilitado en el cliente, el gestor de colas o ambos y cuando TLS1.3 está habilitado tanto en el cliente como en el gestor de colas.

Las tablas siguientes muestran la relación entre los distintos valores de Alias CipherSpec y el resultado esperado. Tabla 76 en la página 439 muestra el comportamiento esperado cuando TLS 1.3 no está habilitado en el cliente, el servidor o ambos. La Tabla 77 en la página 439 muestra el comportamiento esperado cuando TLS 1.3 está habilitado tanto en el cliente como en el servidor. En ambos casos, las CipherSpecs para el cliente se muestran en el eje Y de la tabla, y las CipherSpecs para el servidor se muestran en el eje X de la tabla.

Nota: Donde la entrada indica *Es probable que falle* esto se debe a que, si el TLS específico 1.3 o TLS 1.2 CipherSpec utilizado es el CipherSpec que es más fuerte para el cliente y el gestor de colas, el reconocimiento TLS se resuelve en utilizarlo y, por lo tanto, coincide con el valor SSCIPH del canal.

Tabla 76. Comportamiento esperado cuando TLS 1.3 no está habilitado en el cliente, el servidor o ambos

	Servidor			
Cliente	**TLS específico 1.2 CipherSpec**	**CUALQUIERA**	**ANY_ TLS12**	**ANY_TLS12_ OR_SUPERIOR**
TLS específico 1.2 CipherSpec	Conecta	Conecta	Conecta	Conecta
cualquiera	*Es probable que falle*	Conecta	Conecta	Conecta
ANY_ TLS12	*Es probable que falle*	Conecta	Conecta	Conecta
ANY_TLS12_ OR_SUPERIOR	*Es probable que falle*	Conecta	Conecta	Conecta

Tabla 77. Comportamiento esperado cuando TLS 1.3 está habilitado tanto en el cliente como en el servidor

	Servidor						
Cliente	**TLS específico 1.2 CipherSpec**	**TLS específico 1.3 CipherSpec**	**CUALQUIERA**	**ANY_TLS 12**	**ANY_TLS 13**	**ANY_TLS12_ OR_SUPERIOR**	**ANY_TLS13_ OR_SUPERIOR**
TLS específico 1.2 CipherSpec	Conecta	**Fallos**	Conecta	Conecta	**Fallos**	Conecta	**Fallos**
TLS específico 1.3 CipherSpec	**Fallos**	Conecta	Conecta	**Fallos**	Conecta	Conecta	Conecta
cualquiera	**Fallos**	*Es probable que falle*	Conecta	**Fallos**	Conecta	Conecta	Conecta
ANY_TLS12	*Es probable que falle*	**Fallos**	Conecta	Conecta	**Fallos**	Conecta	**Fallos**
ANY_TLS13	**Fallos**	*Es probable que falle*	Conecta	**Fallos**	Conecta	Conecta	Conecta

Tabla 77. Comportamiento esperado cuando TLS 1.3 está habilitado tanto en el cliente como en el servidor (continuación)

	Servidor						
Cliente	**TLS específico 1.2 CipherSpec**	**TLS específico 1.3 CipherSpec**	**CUALQUIERA**	**ANY_TLS12**	**ANY_TLS13**	**ANY_TLS12_OR_SUPERIOR**	**ANY_TLS13_OR_SUPERIOR**
ANY_TLS12_OR_SUPERIOR	**Fallos**	*Es probable que falle*	Conecta	**Fallos**	Conecta	Conecta	Conecta
ANY_TLS13_OR_SUPERIOR	**Fallos**	*Es probable que falle*	Conecta	**Fallos**	Conecta	Conecta	Conecta

Conceptos relacionados

"Certificados digitales y compatibilidad de CipherSpec en IBM MQ" en la página 45
En este tema se proporciona información sobre cómo elegir las CipherSpecs y los certificados digitales adecuados para su política de seguridad, describiendo la relación entre las CipherSpecs y los certificados digitales en IBM MQ.

"CipherSpecs y CipherSuites" en la página 19
Los protocolos de seguridad criptográficos deben estar de acuerdo con los algoritmos utilizados por una conexión segura. CipherSpecs y CipherSuites definen combinaciones específicas de algoritmos.

"Habilitación de CipherSpecs" en la página 426
Habilite una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC **DEFINE CHANNEL** o el mandato MQSC **ALTER CHANNEL**.

Tareas relacionadas

Migración de configuraciones de seguridad existentes para utilizar la CipherSpec ANY_TLS12_OR_HIGHER

Obtención de información sobre CipherSpecs utilizando IBM MQ Explorer

Puede utilizar IBM MQ Explorer para visualizar descripciones de CipherSpecs.

Utilice el procedimiento siguiente para obtener información acerca de las CipherSpecs que aparecen en la "Habilitación de CipherSpecs" en la página 426:

1. Abra IBM MQ Explorer y expanda la carpeta **Gestores de colas**.
2. Asegúrese de que ha iniciado el gestor de colas.
3. Seleccione el gestor de colas con el que desea trabajar y pulse **Canales**.
4. Pulse con el botón derecho del ratón el canal con el que desee trabajar y seleccione **Propiedades**.
5. Seleccione la página de propiedades **SSL**.
6. Seleccione en la lista la CipherSpec con la que desea trabajar. Se visualiza una descripción en la ventana que hay debajo de la lista.

Alternativas para especificar las CipherSpecs

En aquellas plataformas en las que el sistema operativo proporciona soporte para TLS, el sistema puede dar soporte a nuevas CipherSpecs. Puede especificar una nueva CipherSpec con el parámetro SSLCIPH, pero el valor que suministre dependerá de la plataforma.

Nota: Este apartado no se aplica a sistemas UNIX, Linux o Windows, porque las CipherSpecs se proporcionan con el producto IBM MQ, por lo que no habrá disponibles nuevas CipherSpecs después de la entrega.

En aquellas plataformas en las que el sistema operativo da soporte a TLS, es posible que el sistema dé soporte a nuevas CipherSpecs que no figuran en la "Habilitación de CipherSpecs" en la página 426. Puede especificar una nueva CipherSpec con el parámetro SSLCIPH, pero el valor que suministre dependerá de la plataforma. En todos los casos, la especificación debe corresponder a una TLS CipherSpec que es válida y, también, compatible con la versión de TLS que está ejecutando el sistema.

IBM i

Una serie de dos caracteres que representa un valor hexadecimal.

Si desea más información sobre los valores permitidos, consulte el punto tres en las sección Notas de uso de Establecer información de carácter para una sesión segura.

Atención: No debe especificar valores de cifrado hexadecimal enSSLCIPH, porque no queda claro a partir del valor qué cifrado se utilizará, y la opción de qué protocolo se va a utilizar queda indefinida. El uso de los valores de cifrado hexadecimal puede llevar a errores de discrepancia de CipherSpec.

Puede utilizar el mandato CHGMQMCHL o CRTMQMCHL para especificar el valor; por ejemplo:

```
CRTMQMCHL CHLNAME(' channel name ') SSLCIPH(' hexadecimal value ')
```

También puede utilizar el mandato ALTER QMGR MQSC para establecer el parámetro **SSLCIPH**.

z/OS

Una serie de cuatro caracteres que representa un valor hexadecimal. Los códigos hexadecimales se corresponden con los valores definidos en el protocolo TLS.

Para obtener más información, consulte Definiciones de suite de cifrado donde hay una lista de todas las especificaciones de cifrado TLS 1.0, TLS 1.2y TLS 1.3 soportadas en forma de códigos hexadecimales de 4 dígitos.

Consideraciones sobre los clústeres de IBM MQ

Con los clústeres de IBM MQ, es más seguro utilizar los nombres de CipherSpec de la "Habilitación de CipherSpecs" en la página 426. Si utiliza una especificación alternativa, tenga en cuenta que la especificación puede no ser válida en otras plataformas. Para obtener más información, consulte "SSL/TLS y clústeres" en la página 469.

Especificación de una CipherSpec para un IBM MQ MQI client

Dispone de tres opciones para especificar una CipherSpec para un IBM MQ MQI client.

Estas opciones son las siguientes:

- Utilizar una tabla de definiciones de canal
- Utilizando el campo SSLCipherSpec en la estructura MQCD, en MQCD_VERSION_7 o superior, en una llamada MQCONNX.
- Utilizar Active Directory (en sistemas Windows con soporte para Active Directory)

Especificación de una CipherSuite con IBM MQ classes for Java y IBM MQ classes for JMS

IBM MQ classes for Java y IBM MQ classes for JMS especifican las CipherSuites de forma diferente de otras plataformas.

Para obtener información sobre cómo especificar una CipherSuite con IBM MQ classes for Java, consulte Soporte de TLS (seguridad de la capa de transporte) para Java

Para obtener información sobre cómo especificar una CipherSuite con IBM MQ classes for JMS, consulte Utilización de TLS (seguridad de la capa de transporte) con IBM MQ classes for JMS

Especificación de una CipherSpec para un IBM MQ.NET

En IBM MQ.NET, puede especificar la CipherSpec utilizando la clase MQEnvironment o utilizando MQC.SSL_CIPHER_SPEC_PROPERTY en la tabla hash de las propiedades de conexión.

Para obtener información acerca de cómo especificar una CipherSpec para el cliente .NET no gestionado, consulte Habilitación de TLS para el cliente no gestionado.NET

Para obtener información acerca de cómo especificar una CipherSpec para el cliente .NET gestionado, consulte Soporte de CipherSpec para el cliente .NETgestionado

z/OS Uso de AT-TLS con IBM MQ for z/OS

Application Transparent Transport Layer Security (AT-TLS) proporciona soporte TLS para aplicaciones z/OS sin que dichas aplicaciones tengan que implementar el soporte TLS, o incluso tener en cuenta que se está utilizando TLS. AT-TLS solo está disponible en z/OS.

AT-TLS se puede utilizar con todas las versiones de IBM MQ for z/OS.

Antes de utilizar AT-TLS con IBM MQ for z/OS, asegúrese de que entiende el “restricciones” en la página 444 implicado.

Para utilizar Application Transparent Transport Layer Security , defina sentencias de política que contengan un conjunto de reglas utilizadas por z/OS Communications Server para decidir qué conexiones TCP/IP tienen TLS habilitado de forma transparente.

IBM MQ for z/OS tiene su propia implementación TLS, que requiere que los canales tengan el parámetro SSLCIPH configurado con una CipherSpecsoportada.

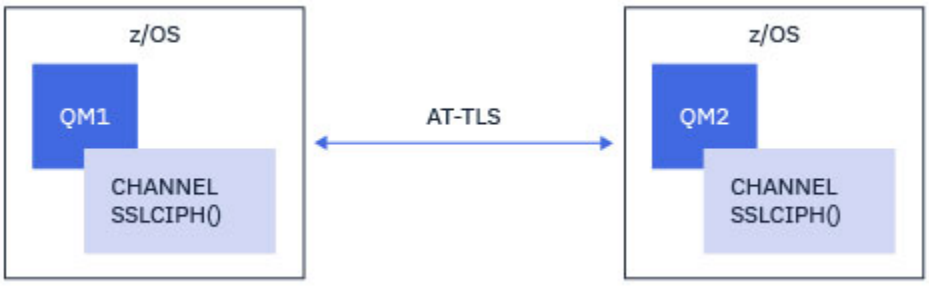
Al decidir habilitar TLS en un canal, el administrador de IBM MQ puede decidir utilizar AT-TLS o IBM MQ TLS. La decisión se suele tomar en función de si se utiliza AT-TLS para otro middleware o debido a implicaciones en el rendimiento. Para obtener una comparación básica del rendimiento de AT-TLS y IBM MQ TLS, consulte MP16: Capacity Planning and Tuning for IBM MQ for z/OS.

Escenarios

El uso de AT-TLS con IBM MQ está soportado en los escenarios siguientes:

Escenario 1

Entre dos gestores de colas de IBM MQ for z/OS en los que ambos lados del canal utilizan AT-TLS. Es decir, ninguno de los canales especifica el atributo SSLCIPH. Este enfoque se puede utilizar con cualquier canal de mensajes.

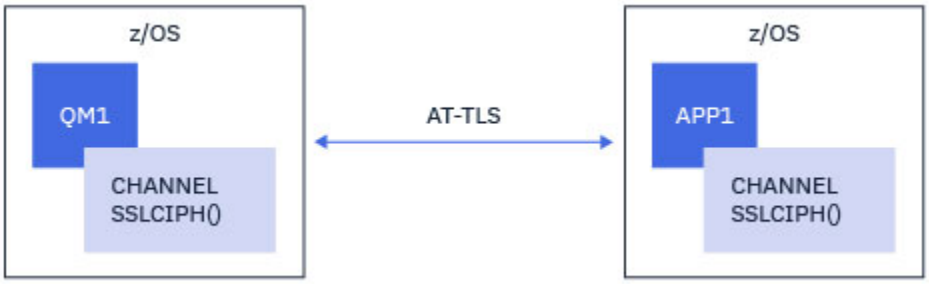


La implementación de este escenario consiste en definir dos políticas AT-TLS, una para cada lado del canal. Estas políticas son las mismas que las utilizadas con el Escenario 3.

Por ejemplo, si el canal se estaba cambiando de utilizar un único CipherSpec a utilizar AT-TLS, el canal de salida utilizaría la política de “Configuración de AT-TLS en un canal de salida a un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 445 y el canal de entrada utilizaría la política de “Configuración de AT-TLS en un canal de entrada desde un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 448.

Escenario 2

Entre un gestor de colas de IBM MQ for z/OS y una aplicación cliente de IBM MQ Java que se ejecuta en z/OS donde ambos lados del canal utilizan AT-TLS. Es decir, ni el canal de conexión de servidor ni el canal de conexión de cliente especifican el atributo SSLCIPH.

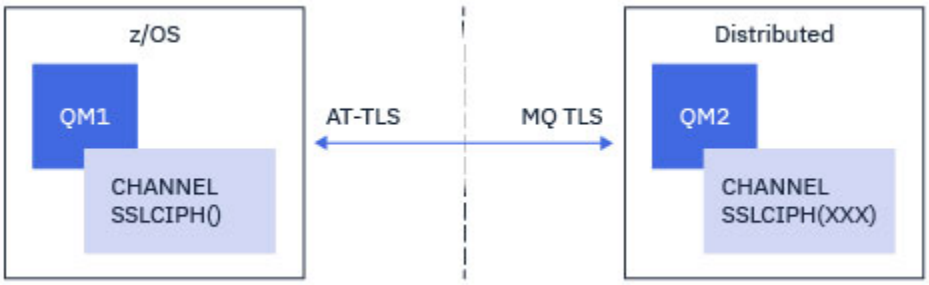


La implementación de este escenario consiste en definir dos políticas AT-TLS, una para cada lado del canal. Estas políticas son las mismas que las utilizadas con el Escenario 3.

Por ejemplo, si el canal se estaba cambiando de utilizar un único CipherSpec a utilizar AT-TLS, el canal de conexión de cliente utilizaría la política de “Configuración de AT-TLS en un canal de salida a un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 445 y el canal de conexión de servidor utilizaría la política de “Configuración de AT-TLS en un canal de entrada desde un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 448.

Escenario 3

Entre un gestor de colas de IBM MQ for z/OS y un gestor de colas que se ejecuta en IBM MQ for Multiplatforms, donde el gestor de colas de IBM MQ for z/OS utiliza AT-TLS y el gestor de colas de IBM MQ for Multiplatforms utiliza IBM MQ TLS. Esto se aplica a todos los tipos de canal de mensajes que no sean clúster emisor y clúster receptor.

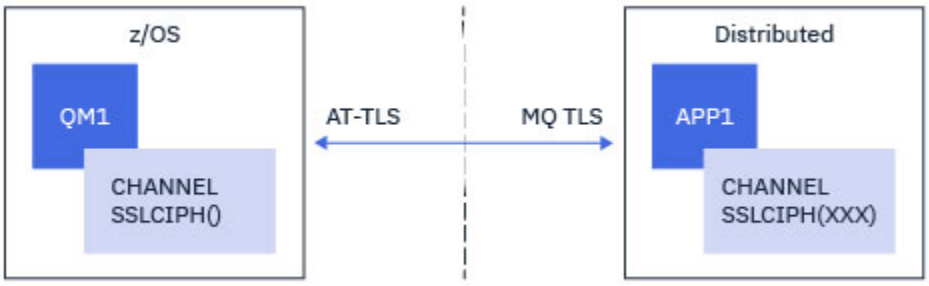


Consulte “Configuración de AT-TLS en un canal de salida a un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 445 para ver un ejemplo de configuración AT-TLS para canales de salida desde el gestor de colas IBM MQ for z/OS al gestor de colas IBM MQ for Multiplatforms , y “Configuración de AT-TLS en un canal de entrada desde un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 448 para ver un ejemplo de configuración AT-TLS para canales de entrada desde el gestor de colas IBM MQ for Multiplatforms al gestor de colas IBM MQ for z/OS .

Se puede utilizar la misma configuración AT-TLS cuando ambos gestores de colas están en z/OS, pero el gestor de colas de la derecha no se ha configurado para utilizar AT-TLS.

Escenario 4

Entre un gestor de colas de IBM MQ for z/OS y una aplicación cliente que se ejecuta en IBM MQ for Multiplatforms, donde el gestor de colas de IBM MQ for z/OS utiliza AT-TLS y la aplicación cliente utiliza IBM MQ TLS especificando el atributo SSLCIPH con una única, denominada CipherSpec.



Este escenario requiere una única política AT-TLS que cumpla los mismos requisitos que los utilizados por un canal de mensajes de entrada; consulte “Configuración de AT-TLS en un canal de entrada desde un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec” en la página 448.

Se puede utilizar la misma configuración AT-TLS cuando la aplicación cliente es una aplicación Java y también se ejecuta en z/OS, pero no se ha configurado para utilizar AT-TLS.

restricciones

IBM MQ for z/OS no tiene en cuenta AT-TLS, por lo tanto, hay varias restricciones que se aplican con los escenarios anteriores:

- AT-TLS en combinación con IBM MQ TLS no funciona con los canales de clúster emisor y clúster receptor.
- Los gestores de colas de IBM MQ for z/OS no son conscientes de que están utilizando AT-TLS y no reciben ninguna información de certificado de su gestor de colas o cliente asociado. Por lo tanto, los atributos siguientes no tienen ningún efecto en el lado z/OS de un canal que utiliza AT-TLS:
  - Los atributos de canal SSLCAUTH y SSLPEER
  - Atributo del gestor de colas SSLRKEYC
  - Los atributos SSLPEERMAP de las reglas CHLAUTH
- El uso de la renegociación de claves secretas TLS requiere que ambos lados del canal utilicen IBM MQ TLS. Por lo tanto, un gestor de colas de IBM MQ for Multiplatforms , o cliente, no debe tener habilitada la renegociación de claves secretas TLS si se conecta a un gestor de colas de IBM MQ for z/OS utilizando AT-TLS.

  Para inhabilitar la renegociación de claves secretas TLS para un gestor de colas, establezca el parámetro SSLRKEYC del gestor de colas en `0`. Para un cliente, establezca el parámetro relevante en `0` en función del tipo de cliente. Para obtener detalles sobre cómo hacerlo, consulte “Restablecimiento de claves secretas SSL y TLS” en la página 452.

Sentencias de configuración AT-TLS

AT-TLS se configura utilizando un conjunto de sentencias. Los utilizados en los escenarios documentados en este tema son:

ReglaTTLS
Especifica un conjunto de criterios para comparar una conexión TCP/IP con una configuración TLS. Esto a su vez hace referencia a los otros tipos de sentencia.

TTLSGroupAction
Especifica si la `TTLSRule` de referencia está habilitada o no.

TTLSEnvironmentAction
Especifica la configuración detallada para el `TTLSRule` de referencia y hace referencia a una serie de otras sentencias.

TTLSKeyringParms
Hace referencia al conjunto de claves que utilizará AT-TLS.

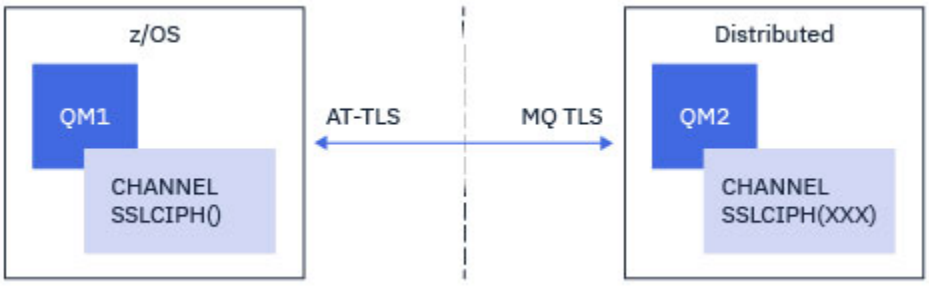
TTLSCipherParms
Define las suites de cifrado que se van a utilizar.

TTLSEnvironmentAdvancedParms
Define qué protocolos TLS o SSL están habilitados.

Atención: Existen otras sentencias de política AT-TLS con AT-TLS que no están documentadas aquí y que se pueden utilizar con IBM MQ en función de las necesidades. Sin embargo, IBM MQ sólo se ha probado con las políticas descritas en este tema.

Configuración de AT-TLS en un canal de salida a un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec

Cómo configurar AT-TLS en un canal de salida desde un gestor de colas de IBM MQ for z/OS a un gestor de colas de IBM MQ for Multiplatforms . En este caso, el canal del gestor de colas z/OS es un canal emisor que no tiene establecido el atributo SSLCIPH, y el canal del gestor de colas que no es dez/OS es un canal receptor con el atributo SSLCIPH establecido en un único, denominado CipherSpec.



En este ejemplo, un par de canales emisor-receptor existente, que utiliza TLS 1.2 TLS_RSA_WITH_AES_256_GCM_SHA384 CipherSpec se va a ajustar para que el canal emisor utilice AT-TLS en lugar de IBM MQ TLS.

Se pueden utilizar otros protocolos TLS y CipherSpecs realizando ajustes menores en la configuración. Otros tipos de canal de mensajes, aparte de los canales de clúster emisor y de clúster receptor, se pueden utilizar sin ningún cambio en la configuración de AT-TLS.

Procedimiento

Paso 1: Detener el canal

Paso 2: Crear y aplicar una política AT-TLS

Debe crear las siguientes sentencias AT-TLS para este escenario:

1. Una sentencia TTLSRule para hacer coincidir las conexiones de salida del espacio de direcciones del iniciador de canal con la dirección IP y el número de puerto del canal receptor de destino. Estos valores deben coincidir con la información utilizada en el CONNAME del canal emisor. Aquí, se

ha incluido un filtrado adicional para que coincida con un nombre de trabajo de iniciador de canal específico.

```
TTLSRule                      CSQ1-TO-REMOTE
{
  LocalAddr                   ALL
  RemoteAddr                  123.456.78.9
  RemotePortRange             1414
  Jobname                     CSQ1CHIN
  Direction                   OUTBOUND
  TTLSGroupActionRef          CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef    CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}
```

La regla anterior coincide con las conexiones que van a la dirección IP 123.456.78.9 en el puerto 1414 del trabajo CSQ1CHIN .

Las opciones de filtrado más avanzadas se describen en TTLSRule.

2. Una sentencia TTLSGroupAction que habilita la regla. TTLSRule hace referencia a TTLSGroupAction utilizando la propiedad **TTLSGroupActionRef** .

```
TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}
```

3. Una sentencia TTLSEnvironmentAction asociada con TTLSRule mediante la propiedad **TTLSEnvironmentActionRef** . Un TTLSEnvironmentAction configura el entorno TLS y especifica qué conjunto de claves se debe utilizar.

```
TTLSEnvironmentAction               CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                     CLIENT
  TTLSKeyringParmsRef               CSQ1-KEYRING
  TTLSCipherParmsRef                CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef   CSQ1-ENVIRONMENT-ADVANCED
}
```

4. Una sentencia TTLSKeyringParms asociada con TTLSEnvironmentAction por la propiedad **TTLSKeyringParmsRef** y define el conjunto de claves utilizado por AT-TLS.

   El conjunto de claves debe contener certificados de confianza del gestor de colas remoto noz/OS . Este conjunto de claves se puede definir de la misma forma que un conjunto de claves utilizado por el iniciador de canal; consulte “Configuración del sistema z/OS para utilizar TLS” en la página 259.

```
TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}
```

5. Una sentencia TTLSCipherParms asociada con TTLSEnvironmentAction mediante la propiedad **TTLSCipherParmsRef** .

   Esta sentencia debe contener un único nombre de suite de cifrado que debe ser el equivalente al nombre de IBM MQ CipherSpec utilizado en el canal receptor de destino.

   Nota: Los nombres de suite de cifrado AT-TLS no coinciden necesariamente con los nombres de IBM MQ CipherSpec . Sin embargo, es posible encontrar el nombre de la suite de cifrado AT-TLS que coincida con un nombre IBM MQ CipherSpec buscando el nombre IBM MQ CipherSpec de la tabla siguiente y haciendo una referencia cruzada a la columna de código de cuatro caracteres con la columna de caracteres expandida de la Tabla 2 en el tema TTLSCipherParms .

Tabla 78. Conversión de códigos de cuatro caracteres a nombres de CipherSpec

Código de cuatro caracteres	Protocolo	Habilitado de forma predeterminada	Nombre de CipherSpec
0001	SSL 3.0	No	NULL_MD5
0002	SSL 3.0	No	NULL_SHA
0003	SSL 3.0	No	RC4_MD5_EXPORT
0004	SSL 3.0	No	RC4_MD5_US
0005	SSL 3.0	No	RC4_SHA_US
0006	SSL 3.0	No	RC2_MD5_EXPORT
0008	SSL 3.0	No	DES_SHA_EXPORT
0009	TLS 1.0	Sí	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	No	TRIPLE_DES_SHA_US
000A	TLS 1.0	Sí	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Sí	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Sí	TLS_RSA_WITH_AES_256_CBC_SHA
003B	TLS 1.2	Sí	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Sí	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Sí	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Sí	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Sí	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Sí	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Sí	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Una sentencia TTLSEnvironmentAdvancedParms está asociada con TTLSEnvironmentAction mediante la propiedad **TTLSEnvironmentAdvancedParmsRef** .

   Esta sentencia se puede utilizar para especificar qué protocolos SSL y TLS están habilitados. Con IBM MQ , solo debe habilitar el protocolo único que coincida con el nombre de suite de cifrado utilizado en la sentencia TTLSCipherParms .

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3           OFF
  TLSv1           OFF
  TLSv1.1         OFF
  SecondaryMap    OFF
  TLSv1.2         ON
  TLSv1.3         OFF
}
```

El conjunto completo de sentencias son las siguientes y se deben aplicar al agente de políticas:

```
TTLSRule                      CSQ1-TO-REMOTE
{
  LocalAddr                   ALL
  RemoteAddr                  123.456.78.9
  RemotePortRange             1414
  Jobname                     CSQ1CHIN
  Direction                   OUTBOUND
  TTLSGroupActionRef          CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef    CSQ1-OUTBOUND-ENVIRONMENT-ACTION
}

TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}

TTLSEnvironmentAction             CSQ1-OUTBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                   CLIENT
  TTLSKeyringParmsRef             CSQ1-KEYRING
  TTLSCipherParmsRef              CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}

TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3          OFF
  TLSv1          OFF
  TLSv1.1        OFF
  SecondaryMap   OFF
  TLSv1.2        ON
  TLSv1.3        OFF
}
```

Paso 3: Eliminar SSLCIPH del canal z/OS

Elimine la CipherSpec del canal z/OS utilizando el mandato siguiente:

```
ALTER CHANNEL(channel-name) CHLTYPE(SDR) SSLCIPH(' ')
```

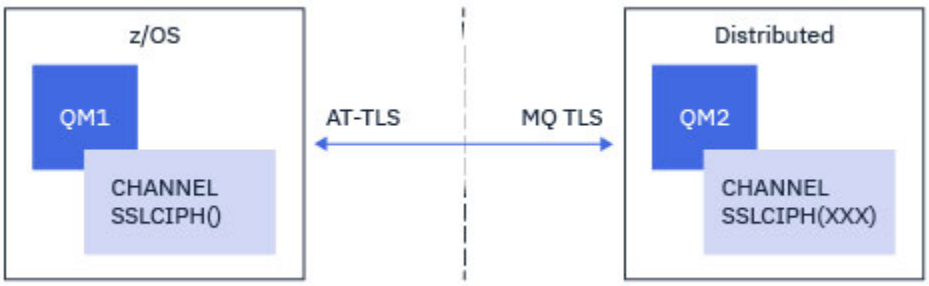
Paso 4: Iniciar el canal

Una vez iniciado el canal, utilizará una combinación de AT-TLS y IBM MQ TLS.

Atención: Las sentencias AT-TLS anteriores son sólo una configuración mínima. Existen otras sentencias de política AT-TLS con AT-TLS que no están documentadas aquí y que se pueden utilizar con IBM MQ en función de las necesidades. Sin embargo, IBM MQ sólo se ha probado con las políticas descritas.

Configuración de AT-TLS en un canal de entrada desde un gestor de colas IBM MQ for Multiplatforms utilizando una única, denominada CipherSpec

Cómo configurar AT-TLS en un canal de entrada desde un gestor de colas de IBM MQ for Multiplatforms a un gestor de colas de IBM MQ for z/OS . En este caso, el canal del gestor de colas z/OS es un canal receptor que no tiene establecido el atributo SSLCIPH, y el canal del gestor de colas noz/OS es un canal emisor con el atributo SSLCIPH establecido en un único, denominado CipherSpec.



En este ejemplo, un par de canales emisor-receptor existente, que utiliza TLS 1.2 TLS_RSA_WITH_AES_256_GCM_SHA384 CipherSpec se va a ajustar para que el canal receptor utilice AT-TLS en lugar de IBM MQ TLS.

Se pueden utilizar otros protocolos TLS y CipherSpecs realizando ajustes menores en la configuración. Otros tipos de canal de mensajes, aparte de los canales de clúster emisor y de clúster receptor, se pueden utilizar sin ningún cambio en la configuración de AT-TLS.

Procedimiento

Paso 1: Detener el canal

Paso 2: Crear y aplicar una política AT-TLS

Debe crear las siguientes sentencias AT-TLS para este escenario:

1. Una sentencia TTLSRule para hacer coincidir las conexiones de entrada con el espacio de direcciones del iniciador de canal desde la dirección IP del canal emisor. Aquí, se ha incluido un filtrado adicional para que coincida con un nombre de trabajo de iniciador de canal específico.

```
TTLSRule                        REMOTE-TO-CSQ1
{
  LocalAddr                     ALL
  LocalPortRange                1414
  RemoteAddr                    123.456.78.9
  Jobname                       CSQ1CHIN
  Direction                     INBOUND
  TTLSGroupActionRef            CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef      CSQ1-INBOUND-ENVIRONMENT-ACTION
}
```

   La regla anterior coincide con las conexiones que entran en el trabajo CSQ1CHIN en el puerto local 1414 desde la dirección IP remota 123.456.78.9.

   Las opciones de filtrado más avanzadas se describen en TTLSRule.

2. Una sentencia TTLSGroupAction que habilita la regla. `TTLSRule` hace referencia a `TTLSGroupAction` utilizando la propiedad **`TTLSGroupActionRef`** .

```
TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}
```

3. Una sentencia TTLSEnvironmentAction se asocia con `TTLSRule` mediante la propiedad **`TTLSEnvironmentActionRef`** . Un `TTLSEnvironmentAction` configura el entorno TLS y especifica qué conjunto de claves se debe utilizar.

```
TTLSEnvironmentAction                 CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                       SERVER
  TTLSKeyringParmsRef                 CSQ1-KEYRING
  TTLSCipherParmsRef                  CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef     CSQ1-ENVIRONMENT-ADVANCED
}
```

AT-TLS proporciona la posibilidad de proporcionar autenticación mutua, que es el equivalente a utilizar el atributo de canal SSLCAUTH. Esto se realiza teniendo una sentencia TTLSEnvironmentAction con un valor **HandshakeRole** de *ServerWithClientAuth* para la sentencia TTLSEnvironmentAction de entrada.

4. Una sentencia TTLSKeyringParms se asocia con TTLSEnvironmentAction mediante la propiedad **TTLSKeyringParmsRef** y define el conjunto de claves utilizado por AT-TLS.

   El conjunto de claves debe contener certificados de confianza del gestor de colas remoto noz/OS . Este conjunto de claves se puede definir de la misma forma que un conjunto de claves utilizado por el iniciador de canal; consulte “Configuración del sistema z/OS para utilizar TLS” en la página 259.

```
TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}
```

5. Una sentencia TTLSCipherParms asociada con TTLSEnvironmentAction mediante la propiedad **TTLSCipherParmsRef** .

   Esta sentencia debe contener un único nombre de suite de cifrado que debe ser el equivalente al nombre de IBM MQ CipherSpec utilizado en el canal emisor remoto.

   Nota: Los nombres de suite de cifrado AT-TLS no coinciden necesariamente con los nombres de IBM MQ CipherSpec . Sin embargo, es posible encontrar el nombre de la suite de cifrado AT-TLS que coincida con un nombre IBM MQ CipherSpec buscando el nombre IBM MQ CipherSpec de la tabla siguiente y haciendo una referencia cruzada a la columna de código de cuatro caracteres con la columna de caracteres expandida de la Tabla 2 en el tema TTLSCipherParms .

Tabla 79. Conversión de códigos de cuatro caracteres a nombres de CipherSpec

Código de cuatro caracteres	Protocolo	Habilitado de forma predeterminada	Nombre de CipherSpec
0001	SSL 3.0	No	NULL_MD5
0002	SSL 3.0	No	NULL_SHA
0003	SSL 3.0	No	RC4_MD5_EXPORT
0004	SSL 3.0	No	RC4_MD5_US
0005	SSL 3.0	No	RC4_SHA_US
0006	SSL 3.0	No	RC2_MD5_EXPORT
0008	SSL 3.0	No	DES_SHA_EXPORT
0009	TLS 1.0	Sí	TLS_RSA_WITH_DES_CBC_SHA
000A	SSL 3.0	No	TRIPLE_DES_SHA_US
000A	TLS 1.0	Sí	TLS_RSA_WITH_3DES_EDE_CBC_SHA
002F	TLS 1.0	Sí	TLS_RSA_WITH_AES_128_CBC_SHA
0035	TLS 1.0	Sí	TLS_RSA_WITH_AES_256_CBC_SHA

Tabla 79. Conversión de códigos de cuatro caracteres a nombres de CipherSpec (continuación)

Código de cuatro caracteres	Protocolo	Habilitado de forma predeterminada	Nombre de CipherSpec
003B	TLS 1.2	Sí	TLS_RSA_WITH_NULL_SHA256
003C	TLS 1.2	Sí	TLS_RSA_WITH_AES_128_CBC_SHA256
003D	TLS 1.2	Sí	TLS_RSA_WITH_AES_256_CBC_SHA256
C023	TLS 1.2	Sí	ECDHE_ECDSA_AES_128_CBC_SHA256
C024	TLS 1.2	Sí	ECDHE_ECDSA_AES_256_CBC_SHA384
C027	TLS 1.2	Sí	ECDHE_RSA_AES_128_CBC_SHA256
C028	TLS 1.2	Sí	ECDHE_RSA_AES_256_CBC_SHA384

```
TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}
```

6. Una sentencia TTLSEnvironmentAdvancedParms está asociada con TTLSEnvironmentAction mediante la propiedad **TTLSEnvironmentAdvancedParmsRef** .

   Esta sentencia se puede utilizar para especificar qué protocolos SSL y TLS están habilitados. Con IBM MQ , solo debe habilitar el protocolo único que coincida con el nombre de suite de cifrado utilizado en la sentencia TTLSCipherParms .

```
TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3           OFF
  TLSv1           OFF
  TLSv1.1         OFF
  SecondaryMap    OFF
  TLSv1.2         ON
  TLSv1.3         OFF
}
```

El conjunto completo de sentencias son las siguientes y se deben aplicar al agente de políticas:

```
TTLSRule                      REMOTE-TO-CSQ1
{
  LocalAddr                   ALL
  LocalPortRange              1414
  RemoteAddr                  123.456.78.9
  Jobname                     CSQ1CHIN
  Direction                   INBOUND
  TTLSGroupActionRef          CSQ1-GROUP-ACTION
  TTLSEnvironmentActionRef    CSQ1-INBOUND-ENVIRONMENT-ACTION
}

TTLSGroupAction             CSQ1-GROUP-ACTION
{
  TTLSEnabled               ON
}

TTLSEnvironmentAction             CSQ1-INBOUND-ENVIRONMENT-ACTION
{
  HandshakeRole                   CLIENT
  TTLSKeyringParmsRef             CSQ1-KEYRING
  TTLSCipherParmsRef              CSQ1-CIPHERPARM
  TTLSEnvironmentAdvancedParmsRef CSQ1-ENVIRONMENT-ADVANCED
}

TTLSKeyringParms            CSQ1-KEYRING
{
  Keyring                   MQCHIN/CSQ1RING
}

TTLSCipherParms             CSQ1-CIPHERPARM
{
  V3CipherSuites            TLS_RSA_WITH_AES_256_GCM_SHA384
}

TTLSEnvironmentAdvancedParms CSQ1-ENVIRONMENT-ADVANCED
{
  SSLv3          OFF
  TLSv1          OFF
  TLSv1.1        OFF
  SecondaryMap   OFF
  TLSv1.2        OFF
  TLSv1.3        ON
}
```

Paso 3: Eliminar SSLCIPH del canal z/OS

Elimine la CipherSpec del canal z/OS utilizando el mandato siguiente:

```
ALTER CHANNEL(channel-name) CHLTYPE(RCVR) SSLCIPH(' ')
```

Paso 4: Iniciar el canal

Una vez iniciado el canal, utilizará una combinación de AT-TLS y IBM MQ TLS.

Atención: Las sentencias AT-TLS anteriores son sólo una configuración mínima. Existen otras sentencias de política AT-TLS con AT-TLS que no están documentadas aquí y que se pueden utilizar con IBM MQ en función de las necesidades. Sin embargo, IBM MQ sólo se ha probado con las políticas descritas.

Restablecimiento de claves secretas SSL y TLS

IBM MQ da soporte al restablecimiento de claves secretas en gestores de colas y clientes.

Las claves secretas se restablecen cuando un número especificado de bytes de datos cifrados han fluido a través del canal. Si las pulsaciones de canal están habilitadas, la clave secreta se restablece antes de que se envíen o reciban datos después de una pulsación de canal.

El valor de restablecimiento de clave siempre se establece inicializando el lado del canal de IBM MQ.

Gestor de colas

Para un gestor de colas, utilice el mandato **ALTER QMGR** con el parámetro **SSLRKEYC** para establecer los valores utilizados durante la renegociación de claves.

En IBM i, utilice **CHGMQM** con el parámetro **SSLRSTCNT**.

Cliente MQI

De forma predeterminada, los clientes MQI no renegocian la clave secreta. Puede hacer que un cliente MQI renegocie la clave de tres formas. En la lista siguiente, los métodos se muestran en orden de prioridad. Si especifica varios valores, se utiliza el valor de prioridad más alto.

1. Utilizar el campo KeyResetCount de la estructura MQSCO en una llamada MQCONNX
2. Utilizar la variable de entorno MQSSLRESET
3. Establecer el atributo SSLKeyResetCount en el archivo de configuración del cliente MQI

Estas variables se pueden establecer en un número entero comprendido entre 0 y 999.999.999, representando el número de bytes no cifrados enviados y recibidos en una conversación TLS antes de que la clave secreta TLS se vuelva a negociar. Especificar un valor 0 indica que las claves secretas TLS no se renegocian nunca. Si especifica una cuenta de restablecimiento de clave secreta TLS entre 1 byte y 32 KB, los canales TLS utilizarán una cuenta de restablecimiento de clave secreta de 32 KB. De esta forma, se evitan restablecimientos de clave excesivos que se producirían para valores de restablecimiento de claves secretas TLS pequeñas.

Si se especifica un valor superior a 0 y las pulsaciones del canal están habilitadas para el canal, la clave secreta también se vuelve a negociar antes de que se envíen o se reciban datos de mensaje tras una pulsación del canal.

El número de bytes hasta la siguiente negociación de la clave secreta se restablece después de cada negociación satisfactoria.

Para obtener todos los detalles de la estructura MQSCO, consulte KeyResetCount (MQLONG). Para obtener información detallada sobre MQSSLRESET, consulte MQSSLRESET. Para obtener más información sobre el uso de TLS en el archivo de configuración de cliente, consulte Stanza SSL del archivo de configuración de cliente.

Java

Para IBM MQ classes for Java, una aplicación puede restablecer la clave secreta en cualquiera de las maneras siguientes:

- Estableciendo el campo sslResetCount en la clase MQEnvironment.
- Estableciendo la propiedad de entorno MQC.SSL_RESET_COUNT_PROPERTY en un objeto Hashtable. A continuación, la aplicación asigna la tabla hash al campo `properties` en la clase MQEnvironment o pasa la tabla hash a un objeto MQQueueManager de su constructor.

Si la aplicación utiliza más de uno de estos métodos, se aplican las reglas de prioridad habituales. Consulte Clase com.ibm.mq.MQEnvironment para las reglas de prioridad.

El valor del campo sslResetCount o de la propiedad de entorno MQC.SSL_RESET_COUNT_PROPERTY representa el número total de bytes enviados y recibidos por el código de cliente IBM MQ classes for Java antes de que se renegocie la clave secreta. El número de bytes enviados es el número antes del cifrado y el número de bytes recibidos es el número después del cifrado. El número de bytes incluye también la información de control enviada y recibida por el cliente IBM MQ classes for Java.

Si la cuenta de restablecimiento es cero, que es el valor predeterminado, la clave secreta nunca se renegocia. La cuenta de restablecimiento se ignora si no se especifica ninguna CipherSuite.

JMS

Para IBM MQ classes for JMS, la propiedad SSLRESETCOUNT representa el número total de bytes enviados y recibidos por una conexión antes de renegociar la clave secreta que se utiliza para el cifrado. El número de bytes enviados es el número antes del cifrado y el número de bytes recibidos es el número después del cifrado. El número de bytes también incluye información de control enviada y recibida por IBM MQ classes for JMS. Por ejemplo, para configurar un objeto ConnectionFactory que se puede utilizar para crear una conexión a través de un canal MQI habilitado para TLS con una clave secreta que se renegocia después de que hayan fluido 4 MB de datos, emita el mandato siguiente en JMSAdmin:

```
ALTER CF(my.cf) SSLRESETCOUNT(4194304)
```

Si el valor de SSLRESETCOUNT es cero, que es el valor predeterminado, la clave secreta nunca se renegocia. La propiedad SSLRESETCOUNT se ignora si SSLCIPHERSUITE no está establecido.

.NET

Para los clientes no gestionados .NET, la propiedad de entero SSLKeyResetCount indica el número de bytes no cifrados enviados y recibidos en una conversación TLS antes de que se renegocie la clave secreta.

Para obtener información sobre el uso de las propiedades del objeto en IBM MQ classes for .NET, consulte Obtención y establecimiento de valores de atributo.

En los clientes gestionados de .NET, la clase SSLStream no soporta restablecimiento/negociación de clave secreta. Sin embargo, para ser coherente con otros clientes IBM MQ, el cliente gestionadoIBM MQ de .NET permite que las aplicaciones cliente estblezcan SSLKeyResetCount. Para obtener más información, consulte Restablecimiento o renegociación de una clave secreta.

XMS .NET

En clientes no gestionados de .NET XMS, consulte Conexiones seguras con un gestor de colas de IBM MQ.

Referencia relacionada
ALTER QMGR
DISPLAY QMGR
Cambiar gestor de colas de mensajes (CHGMQM)
Visualizar gestor de colas de mensajes (DSPMQM)

Implementación de confidencialidad en programas de salida de usuario

Implementación de confidencialidad en salidas de seguridad

Las salidas de seguridad pueden jugar un papel en el servicio de confidencialidad, generando y distribuyendo la clave simétrica para cifrar y descifrar los datos que fluyen en el canal. Una técnica común para hacerlo utiliza la tecnología PKI.

Una salida de seguridad genera un valor de datos aleatorio, lo cifra con la clave pública del gestor de colas o usuario al que representa la salida de seguridad del asociado y envía los datos cifrados a su asociado en un mensaje de seguridad. La salida de seguridad del asociado descifra el valor de datos aleatorio con la clave privada de gestor de claves o usuario al que representa. Ahora cada salida de seguridad puede utilizar el valor de datos aleatorio para obtener la clave simétrica, independientemente de la otra, utilizando un algoritmo que ambas conocen. Como alternativa, pueden utilizar el valor de datos aleatorio como clave.

Si la primera salida de seguridad aún no ha autenticado a su asociado, el siguiente mensaje de seguridad que envía el asociado puede contener un valor esperado cifrado con la clave simétrica. Ahora la primera salida de seguridad puede autenticar a su asociado comprobando que la salida de seguridad del asociado ha podido cifrar correctamente el valor esperado.

Las salidas de seguridad también pueden utilizar esta oportunidad para acordar el algoritmo para cifrar y descifrar los datos que fluyen en el canal, en el caso de que se pueda utilizar más de un algoritmo.

Implementación de confidencialidad en salidas de mensajes

Una salida de mensajes del extremo emisor de un canal puede cifrar los datos de aplicación en un mensaje y otra salida de mensajes en el extremo receptor del canal puede descifrar los datos. Por razones de rendimiento, para esta finalidad se utiliza normalmente un algoritmo de clave simétrica. Para obtener más información sobre cómo se puede generar y distribuir la clave simétrica, consulte el apartado "Implementación de confidencialidad en programas de salida de usuario" en la página 454.

Las cabeceras de un mensaje, como la cabecera de la cola de transmisión MQXQH, que incluye el descriptor de mensaje incorporado, no se pueden cifrar mediante una salida de mensajes. Esto se debe a que la conversión de datos de las cabeceras de mensajes se lleva a cabo después de que se llame a la salida de mensajes en el extremo emisor o antes de que se llame a la salida de mensajes en el extremo receptor. Si las cabeceras están cifradas, la conversión de datos da error y el canal se detiene.

Implementación de confidencialidad en salidas de emisión y recepción

Las salidas de emisión y recepción se pueden utilizar para cifrar y descifrar los datos que fluyen en un canal. Resultan más adecuadas que las salidas de mensajes para proporcionar este servicio por los siguientes motivos:

- En un canal de mensajes, las cabeceras de mensajes se pueden cifrar, al igual que los datos de aplicación de los mensajes.
- Las salidas de emisión y recepción se pueden utilizar tanto en canales MQI como en canales de mensajes. Los parámetros de las llamadas MQI pueden contener datos que dependan de la aplicación que se tengan que proteger mientras fluyen en un canal MQI. Por lo tanto, puede utilizar las mismas salidas de emisión y recepción en ambos tipos de canales.

Implementación de confidencialidad en la salida de API y la salida cruzada de API

Una salida de API o salida cruzada de API puede cifrar los datos de aplicación de un mensaje cuando la aplicación emisora transfiere el mensaje y una segunda salida puede descifrarlos cuando la aplicación receptora recupera el mensaje. Por razones de rendimiento, para esta finalidad se utiliza normalmente un algoritmo de clave simétrica. No obstante, a nivel de aplicación, en el que muchos usuarios se pueden estar enviando mensajes, el problema es garantizar que sólo el receptor al que va destinado un mensaje pueda descifrar el mensaje. Una solución es utilizar una clave simétrica diferente para cada par de usuarios que se envían mensajes entre sí. Pero administrar esta solución puede resultar difícil y requerir mucho tiempo, sobretodo si los usuarios pertenecen a organizaciones diferentes. Un método estándar de resolver este problema es el que se conoce como *sobre digital* y utiliza la tecnología PKI.

Cuando una aplicación transfiere un mensaje a una cola, una salida de API o salida cruzada de API genera una clave simétrica aleatoria y utiliza la clave para cifrar los datos de aplicación incluidos en el mensaje. La salida cifra la clave simétrica con la clave pública del receptor al que va destinado. A continuación, sustituye los datos de aplicación del mensaje por los datos de aplicación cifrados y la clave simétrica cifrada. De este modo, solamente el receptor al que va destinado puede descifrar la clave simétrica y, por lo tanto, los datos de aplicación. Si un mensaje cifrado va destinado a más de un receptor, la salida puede cifrar una copia de la clave simétrica para cada receptor al que va destinado.

Si se dispone de algoritmos diferentes para cifrar y descifrar los datos de aplicación, la salida puede incluir el nombre del algoritmo que ha utilizado.

z/OS V 9.1.4 Confidencialidad de los datos en reposo en IBM MQ for z/OS con cifrado de conjunto de datos

IBM MQ for z/OS puede proteger los datos de cliente y de configuración escribiendo los datos en los conjuntos de datos de registro activo, los conjuntos de datos de registro de archivado, conjuntos de

páginas conjuntos de datos de rutina de arranque (BSDS) y V 9.1.5 conjuntos de datos de mensajes compartidos (SMDS).

z/OS proporciona un cifrado eficaz y basado en políticas de conjuntos de datos. IBM MQ for z/OS da soporte al cifrado de conjunto de datos de z/OS para:

- Conjuntos de datos de registro activo; consulte la nota “1” en la página 456
- Conjuntos de datos de registro de archivado; consulte la nota “2” en la página 456
- Conjuntos de páginas; consulte la nota “1” en la página 456
- BSDS; consulte la nota “2” en la página 456
- Conjuntos de datos CSQINP*; consulte la nota “2” en la página 456
- V 9.1.5 SMDS; consulte la nota “3” en la página 456

Esto proporciona la confidencialidad de los datos en reposo en un gestor de colas z/OS individual.

Notas:

1. A partir de IBM MQ 9.1.4, IBM MQ for z/OS da soporte al cifrado de conjuntos de datos de z/OS para registros activos y conjuntos de páginas.
2. Se da soporte al cifrado de conjuntos de datos para registros de archivado, BSDS y conjuntos de datos CSQINP* en todas las versiones de IBM MQ for z/OS.
3. V 9.1.5 A partir de IBM MQ 9.1.5, IBM MQ for z/OS da soporte al cifrado de conjuntos de datos de z/OS para SMDS.
4. IBM MQ Advanced Message Security proporciona un mecanismo alternativo de protección de datos en reposo. Además, AMS también protege los datos en la memoria y en movimiento

Consulte Utilización de las mejoras de cifrado de conjuntos de datos de z/OS para obtener más información sobre el cifrado del conjunto de datos de z/OS.

La configuración del cifrado del conjunto de datos de z/OS está fuera del control de IBM MQ for z/OS. Los valores de cifrado se aplican cuando se crea el conjunto de datos.

Esto significa que cualquier conjunto de datos existente se debe volver a crear antes de poder utilizar una nueva política de cifrado de conjunto de datos.

IBM MQ for z/OS se puede ejecutar con una combinación de conjuntos de datos cifrados y no cifrados, pero una configuración estándar cifraría todos los conjuntos de datos utilizados, o ninguno de ellos.

z/OS V 9.1.4 Visión general de los pasos para cifrar un conjunto de datos de IBM MQ for z/OS

Cómo se cifra un conjunto de datos de IBM MQ for z/OS.

Antes de empezar

Debe asegurarse de que ha configurado el cifrado de conjunto de datos de z/OS correctamente en la empresa. Si está configurando el cifrado de conjuntos de datos en un grupo de compartición de colas, debe configurar el cifrado de conjunto de datos de z/OS para el compartimiento de datos.

Nota: Un conjunto de datos cifrado de z/OS debe ser un conjunto de datos de formato ampliado.

Procedimiento

1. Configure la clave de cifrado y `key-label` en RACF para utilizarlos para cifrar el conjunto de datos.
2. Cree un perfil para `key-label` en la clase CSFKEYS de RACF.
3. Otorgue acceso de lectura (READ) al ID de usuario del gestor de colas y a los demás ID de usuario que necesiten acceder a los datos cifrados.

Esto podría incluir los ID de usuario que se utilizan para ejecutar programas de utilidad de impresión en el conjunto de datos. Por ejemplo, el usuario que ejecuta CSQUTIL SCOPY debería descifrar el conjunto de páginas relevantes.

4. Asocie el cifrado `key-label` con el nombre de conjunto de datos.

   Para ello, utilice una clase de datos SMS o un segmento DFP de RACF, para el nombre del conjunto de datos o el calificador de alto nivel.

   También puede asociar `key-label` con el conjunto de datos cuando se asigna el conjunto de datos.
5. Cambie el nombre de cualquier conjunto de datos existente utilizando IDCAMS ALTER.
6. Vuelva a asignar el conjunto de datos con los atributos adecuados.
7. Copie el contenido del conjunto de datos renombrado al nuevo conjunto de datos utilizando IDCAMS REPRO.

   Los datos se cifran mediante la acción de copiarlos en el conjunto de datos.
8. Repita los pasos “4” en la página 457 a “6” en la página 457 para cualquier otro conjunto de datos que sea necesario cifrar.

z/OS V 9.1.4 Ejemplo de cómo cifrar registros activos del gestor de colas

Los temas siguientes le guían a través del proceso de habilitación del cifrado del conjunto de datos en los registros activos existentes.

Nota: El proceso para otros conjuntos de datos es similar al de los registros activos.

En este ejemplo:

- El gestor de colas CSQ1 se ejecuta con el usuario QMCSQ1 y tiene los conjuntos de datos de registro activos CSQ1.LOGS.LOGCOPY1.DS001, CSQ1.LOGS.LOGCOPY1.DS002, etc.
- El entorno de hardware y software puede utilizar el cifrado de conjuntos de datos de z/OS
- RACF se utiliza como SAF
- El gestor de colas se ha detenido

Lleve a cabo el procedimiento en el orden siguiente:

1. “Configuración de la clave de cifrado del conjunto de datos para el gestor de colas” en la página 457
2. “Configuración del cifrado del conjunto de datos para los conjuntos de datos de registro” en la página 458

z/OS V 9.1.4 Configuración de la clave de cifrado del conjunto de datos para el gestor de colas

Cómo configurar una clave de cifrado de conjunto de datos para un gestor de colas.

Acerca de esta tarea

Esta tarea es un requisito previo para “Configuración del cifrado del conjunto de datos para los conjuntos de datos de registro” en la página 458.

Procedimiento

1. Configure una clave DATA de cifrado de bits AES-256 con una etiqueta, por ejemplo, CSQ1DSKY, utilizando el programa de utilidad de generador de claves (KGUP) z/OS https://www.ibm.com/docs/en/zos/2.5.0?topic=keys-key-generator-utility-program.
2. Defina el perfil CSFKEYS de RACF para la clave de cifrado CSQ1DSKY, emitiendo el mandato siguiente:

```
RDEFINE CSFKEYS CSQ1DSKY UACC(NONE)
```

3. Configure el segmento ICSF del perfil para permitir que la clave se utilice como una clave protegida, emitiendo el mandato siguiente:

```
RALTER CSFKEYS CSQ1DSKY ICSF(SYMCPACFWRAP(YES) SYMCPACFRET(YES))
```

4. Permita que el gestor de colas utilice la clave de cifrado dando a QMCSQ1 acceso READ al perfil, emitiendo el mandato siguiente:

```
PERMIT CSQ1DSKY CLASS(CSFKEYS) ID(QMCSQ1) ACCESS(READ)
```

   Otorgue el mismo acceso a cualquier usuario administrativo que deba leer o escribir el conjunto de datos cifrado.
5. Renueve la clase CSFKEYS emitiendo el siguiente mandato:

```
SETROPTS RACLIST(CSFKEYS) REFRESH
```

Qué hacer a continuación

Configure el cifrado de conjunto de datos para los conjuntos de datos tal como se describe en "Configuración del cifrado del conjunto de datos para los conjuntos de datos de registro" en la página 458

z/OS V 9.1.4 Configuración del cifrado del conjunto de datos para los conjuntos de datos de registro

Cómo se configura el cifrado en los conjuntos de datos de registro.

Antes de empezar

Asegúrese de que ha hecho lo siguiente:

Ha leído Visión general de los pasos para cifrar un conjunto de datos de IBM MQ for z/OS y ha llevado a cabo el procedimiento en
"Configuración de la clave de cifrado del conjunto de datos para el gestor de colas" en la página 457

Acerca de esta tarea

Este método utiliza el segmento DFP de un perfil genérico de RACF, de modo que puede utilizar la clave de cifrado para todos los nuevos conjuntos de datos que coinciden con el perfil.

Como alternativa, puede configurar y utilizar una clase de datos SMS, o la etiqueta de clave se puede especificar directamente al asignar el conjunto de datos.

Como se ha descrito anteriormente, en este ejemplo, el gestor de colas CSQ1 se ejecuta con el usuario QMCSQ1, y tiene los conjuntos de datos de registro activo CSQ1.LOGS.LOGCOPY1.DS001, CSQ1.LOGS.LOGCOPY1.DS002, etc.

Procedimiento

1. Cree el perfil genérico si no existe, emitiendo el mandato siguiente:

```
ADDSD 'CSQ1.LOGS.*' UACC(NONE)
```

2. Permita que el usuario del gestor de colas modifique el acceso en el perfil, emitiendo el mandato siguiente:

```
PERMIT 'CSQ1.LOGS.*' ID(QMCSQ1) ACCESS(ALTER)
```

   Asimismo, permita el acceso adecuado necesario para cualquier usuario administrativo.
3. Añada el segmento DFP con la etiqueta de clave de cifrado emitiendo el mandato siguiente:

```
ALTDSD 'CSQ1.LOGS.*' DFP(RESOWNER(QMCSQ1) DATAKEY(CSQ1DSKY))
```

Nota: Debe utilizar la misma clave de cifrado que ha utilizado en la configuración de la clave de cifrado de conjunto de datos para el gestor de colas.

4. Renueve los perfiles de conjunto de datos genéricos emitiendo el mandato siguiente:

```
SETROPTS GENERIC(DATASET) REFRESH
```

5. Cambie el nombre de cada conjunto de datos de registro a una copia de seguridad y, a continuación, vuelva a crear y restaure los datos utilizando IDCAMS. El fragmento de JCL siguiente convierte CSQ1.LOGS.LOGCOPY1.DS001:

   a) Cambie el nombre del conjunto de datos a una copia de seguridad

```
//RENAME   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RENAME DATASET TO BACKUP                                    */
  /*-------------------------------------------------------------*/
  ALTER 'CSQ1.LOGS.LOGCOPY1.DS001'                  -
        NEWNAME('CSQ1.BAK.LOGS.LOGCOPY1.DS001')
```

   b) Vuelva a definir el conjunto de datos.

   El nuevo conjunto de datos se cifrará debido al perfil RACF.

   Nota: Sustituya ++EXTDCLASS++ con el nombre de la clase de datos de formato ampliado que desea utilizar para el conjunto de datos.

```
//REDEFINE EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* REDEFINE THE DATASET                                        */
  /*-------------------------------------------------------------*/
  DEFINE CLUSTER                                    -
         (NAME(CSQ1.LOGS.LOGCOPY1.DS001)            -
         LINEAR                                     -
         SHAREOPTIONS(2 3)                          -
         MODEL(CSQ1.BAK.LOGS.LOGCOPY1.DS001)        -
         DATACLAS(++EXTDCLASS++))
```

   c) Copie los datos de la copia de seguridad en el conjunto de datos recreado.

   Este paso cifra los datos:

```
//RESTORE  EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RESTORE DATA INTO ENCRYPTED LOG                             */
  /*-------------------------------------------------------------*/
  REPRO INDATASET(CSQ1.BAK.LOGS.LOGCOPY1.DS001)     -
        OUTDATASET(CSQ1.LOGS.LOGCOPY1.DS001)
```

Qué hacer a continuación

Repita el paso “5” en la página 459 para todos los conjuntos de datos de registro activo.

Sólo se requiere una única clave de cifrado, y todos los conjuntos de datos se pueden asociar con la misma etiqueta de clave.

Reinicie el gestor de colas CSQ1. Utilice la salida del mandato DISPLAY LOG para verificar que los conjuntos de datos de registro se han cifrado.

z/OS V 9.1.4 Consideraciones sobre el cifrado de conjuntos de datos de z/OS en un grupo de compartición de colas

Cada gestor de colas de un grupo de compartición de colas (QSG) debe poder leer los registros, BSDS V 9.1.5 y los conjuntos de datos de mensajes compartidos (SMDS), de todos los demás gestores de colas del QSG.

Esto significa que cada sistema en el que un miembro del QSG se puede ejecutar debe cumplir los requisitos para el cifrado de conjuntos de datos de z/OS y que todas las etiquetas de clave y claves de cifrado utilizadas para proteger los conjuntos de datos para cada gestor de colas del QSG deben estar disponibles en cada sistema.

Un gestor de colas anterior a IBM MQ for z/OS 9.1.3 no puede acceder a un conjunto de datos de registro activo cifrado.

V 9.1.5 Un gestor de colas anterior a IBM MQ for z/OS 9.1.3 no puede acceder a un SMDS cifrado.

V 9.1.5 Antes de utilizar el cifrado de conjuntos de datos de z/OS , debe migrar todos los gestores de colas de un QSG al menos a IBM MQ for z/OS 9.1.3.

Si un gestor de colas de un QSG se inicia con cualquier conjunto de datos de registro activo cifrado, y cualquier otro gestor de colas del QSG se ha iniciado, pero no se ha iniciado por última vez con una versión de IBM MQ for z/OS que dé soporte a registros activos cifrados, el gestor de colas con el registro activo cifrado termina de forma anómala con el código de terminación anómala 5C6-00F50033.

V 9.1.5 Puede convertir un QSG para que utilice registros activos cifrados y SMDS sin una interrupción completa, haciendo lo siguiente:

1. Migrando a su vez cada gestor de colas al menos a IBM MQ 9.1.5.
2. Convirtiendo a su vez los registros activos en conjuntos de datos cifrados para cada gestor de colas. Esto requiere que el gestor de colas se cierre y luego se reinicie.

   Al mismo tiempo, es probable que los conjuntos de páginas y los registros de archivado se habiliten también para los conjuntos de datos cifrados, pero esto no afecta a la migración de QSG.

   El procedimiento para convertir cada conjunto de datos se describe en “Ejemplo de cómo cifrar registros activos del gestor de colas” en la página 457
3. Convirtiendo a su vez el SMDS en conjuntos de datos cifrados para cada estructura CF individual haciendo lo siguiente:
   a. Emitiendo el mandato RESET SMDS (*) ACCESS (DISABLED) CFSTRUCT (structure-name) para suspender el acceso del gestor de colas al SMDS.

      Tenga en cuenta que durante este tiempo los datos de las colas compartidas asociadas con el SMDS no están disponibles temporalmente.
   b. Convirtiendo cada conjunto de datos que conforma el SMDS en conjuntos de datos cifrados, utilizando el procedimiento que se describe en “Ejemplo de cómo cifrar registros activos del gestor de colas” en la página 457.
   c. Emitiendo el mandato RESET SMDS (*) ACCESS (ENABLED) CFSTRUCT (structure-name) para reanudar el acceso del gestor de colas al SMDS.

Atención: Debe cerrar el gestor de colas correctamente antes de convertir los registros, y es posible que la recuperación de la estructura del recurso de acoplamiento no sea posible durante la conversión, ya que los conjuntos de datos de registro activo no estarán disponibles temporalmente.

z/OS V 9.1.4 Consideraciones sobre la migración al utilizar el cifrado de conjuntos de datos de z/OS

Debe tener en cuenta lo siguiente al realizar la migración a una versión anterior de un gestor de colas con uno o más conjuntos de datos cifrados.

El cifrado de conjuntos de datos de z/OS está soportado en los siguientes conjuntos de datos de IBM MQ for z/OS:

- Conjuntos de datos de registro activo
- Conjuntos de datos de registro de archivado
- Conjuntos de páginas
- BSDS
- V 9.1.5 SMDS
- Conjuntos de datos CSQINP*

No hay consideraciones sobre la migración a versiones anteriores para BSDS, el registro de archivado o los conjuntos de datos CSINP*.

Sin embargo, hay consideraciones para

- V 9.1.5 SMDS
- Conjunto de páginas y
- Registro activo

conjuntos de datos, ya que el uso de estos con el cifrado de conjuntos de datos z/OS no está soportado en IBM MQ for z/OS 9.1.0y en releases anteriores de soporte a largo plazo.

Antes de la migración a versiones anteriores, todas las políticas de cifrado para el conjunto de páginas V 9.1.5 SMDS y los conjuntos de datos de registro activo se deben eliminar y los datos se deben descifrar. Este proceso se describe en "Eliminación del cifrado de conjunto de datos de un conjunto de datos" en la página 461.

Atención: Si el gestor de colas que se migrará a una versión anterior forma parte de un grupo de compartición de colas (QSG), lea primero la sección "Consideraciones sobre los grupos de compartición de colas" en la página 463.

Eliminación del cifrado de conjunto de datos de un conjunto de datos

En este ejemplo se describe cómo eliminar el cifrado de conjunto de datos del conjunto de datos de registro CSQ1.LOGS.LOGCOPY1.DS001. Puede utilizar un proceso equivalente para V 9.1.5 SMDS y conjuntos de páginas.

En el ejemplo se supone que:

- RACF es SAF
- Se ha detenido el gestor de colas que utiliza el conjunto de datos
- La etiqueta de clave de cifrado se ha asociado con el perfil RACF genérico CSQ1.LOGS.*

Lleve a cabo el procedimiento siguiente:

1. Copie los datos del conjunto de datos en un conjunto de datos de copia de seguridad.
   a. Defina un conjunto de datos de copia de seguridad que no esté asociado a una etiqueta de clave de cifrado.

Nota: Sustituya ++EXTDCLASS++ con el nombre de la clase de datos de formato ampliado que desea utilizar para el conjunto de datos.

```
//DEFINE EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* DEFINE UNENCRYPTED DATA SET                                 */
  /*-------------------------------------------------------------*/
  DEFINE CLUSTER                                    -
         (NAME(CSQ1.BAK.LOGS.LOGCOPY1.DS001)        -
         LINEAR                                     -
         SHAREOPTIONS(2 3)                          -
         MODEL(CSQ1.LOGS.LOGCOPY1.DS001)            -
         DATACLAS(++EXTDCLASS++))
/*
```

b. Copie los datos del conjunto de datos original en la copia de seguridad. Este paso descifra los datos.

```
//COPY     EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* COPY DATA INTO UNENCRYPTED DATA SET                         */
  /*-------------------------------------------------------------*/
  REPRO INDATASET(CSQ1.LOGS.LOGCOPY1.DS001)          -
        OUTDATASET(CSQ1.BAK.LOGS.LOGCOPY1.DS001)
/*
```

c. Suprima el conjunto de datos original

```
//DELETE   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* DELETE ORIGINAL                                             */
  /*-------------------------------------------------------------*/
  DELETE ('CSQ1.LOGS.LOGCOPY1.DS001')
/*
```

d. Cambie el nombre de la copia de seguridad al nombre del conjunto de datos original. Los datos siguen estando sin cifrar

```
//RENAME   EXEC PGM=IDCAMS,REGION=0M
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  /*-------------------------------------------------------------*/
  /* RENAME UNENCRYPTED DATA SET                                 */
  /*-------------------------------------------------------------*/
  ALTER CSQ1.BAK.LOGS.LOGCOPY1.DS001'                -
         NEWNAME('CSQ1.LOGS.LOGCOPY1.DS001)
  ALTER 'CSQ1.BAK.LOGS.LOGCOPY1.DS001.*'              -
         NEWNAME('CSQ1.LOGS.LOGCOPY1.DS001.*')
/*
```

2. Opcionalmente, repita este proceso para otros conjuntos de datos que tengan una etiqueta de clave de cifrado asociada a ellos a través de CSQ1.LOGS.* perfil genérico.
3. Opcionalmente, si todos los conjuntos de datos asociados con CSQ1.LOGS.* perfil genérico se han descifrado, elimine el DATAKEY asociado con el perfil genérico emitiendo el mandato siguiente

```
ALTDSD 'CSQ1.LOGS.*' DFP(RESOWNER(QMCSQ1) DATAKEY(CSQ1DSKY))
```

4. Renueve los perfiles de conjunto de datos genéricos emitiendo el mandato siguiente:

```
SETROPTS GENERIC(DATASET) REFRESH
```

5. Reinicie el gestor de colas.

6. Si la clave de cifrado ya no es necesaria, suprímala y suprima su perfil RACF asociado de la clase CSFKEYS.

Consideraciones sobre los grupos de compartición de colas

Si un gestor de colas que forma parte de un grupo de compartición de colas se va a migrar a una versión anterior de IBM MQ for z/OS que no da soporte al cifrado de conjuntos de datos, es necesario eliminar las políticas de cifrado de conjunto de datos y descifrar los datos de todos los conjuntos de datos de registro activo V 9.1.5 y SMDS de todos los gestores de colas en el QSG.

Esto es aplicable tanto si se migra a una versión anterior un único miembro del QSG como si se migran todos.

Puede lograr la eliminación de políticas de cifrado y el descifrado de datos, sin una interrupción completa de QSG haciendo lo siguiente:

1. Concluyendo a su vez cada gestor de colas en el QSG, eliminando las políticas de cifrado y descifrando los datos de sus registros activos, utilizando el proceso descrito en "Eliminación del cifrado de conjunto de datos de un conjunto de datos" en la página 461.

   Si el gestor de colas se va a migrar a una versión anterior, su conjunto de páginas también debe descifrarse en este momento. A continuación, reinicie el gestor de colas.

2. V 9.1.5 Eliminando a su vez las políticas de cifrado y descifrando los datos para el SMDS de cada estructura CF individual, haciendo lo siguiente:

   a. Emitiendo el mandato

   ```
   RESET SMDS(*) ACCESS(DISABLED) CFSTRUCT(structure-name)
   ```

   para suspender el acceso del gestor de colas al SMDS. Durante este tiempo los datos de las colas compartidas asociadas con el SMDS no están disponibles temporalmente.

   b. Siguiendo el proceso en "Eliminación del cifrado de conjunto de datos de un conjunto de datos" en la página 461 para cada conjunto de datos que compone el SMDS.

   c. Emitiendo el mandato

   ```
   RESET SMDS(*) ACCESS(ENABLED) CFSTRUCT(structure-name)
   ```

   para reanudar el acceso del gestor de colas al SMDS.

Utilizando el cifrado de conjuntos de datos de z/OS con un gestor de colas que no le da soporte

Si accidentalmente vuelve a migrar un gestor de colas a una versión anterior de IBM MQ for z/OS que no da soporte al cifrado de conjuntos de datos, y olvida eliminar las políticas de cifrado y descifrar los datos, se produce un error cuando el gestor de colas intenta acceder al conjunto de datos.

El error depende del tipo de conjunto de datos y se muestra en la tabla siguiente.

Nota: Si se producen uno o más de estos errores, debe seguir los procesos que se describen en "Eliminación del cifrado de conjunto de datos de un conjunto de datos" en la página 461 para el conjunto de datos afectado. Estos se pueden realizar sin cambiar la versión de IBM MQ for z/OS.

Conjunto de datos	Error si el gestor de colas no da soporte al cifrado de conjuntos de datos de z/OS
Conjunto de páginas 0	Terminación anómala 5C6-00C91400 durante el inicio del gestor de colas
Conjuntos de páginas 1-99	MQRC 2193 "Error de conjunto de páginas" al acceder al conjunto de páginas, por ejemplo, en MQPUT

Conjunto de datos	Error si el gestor de colas no da soporte al cifrado de conjuntos de datos de z/OS
Registro activo	Terminación anómala 5C6-00E80084 durante el inicio del gestor de colas
V 9.1.5 SMDS	El mensaje IEC161I-122 registrado “El conjunto de datos tiene un KEYLABEL, pero el usuario no ha especificado que la aplicación pueda gestionar el cifrado”. SMDS ha marcado AVAIL(ERROR).

Integridad de datos de mensajes

Para mantener la integridad de los datos, puede utilizar varios tipos de programas de salida de usuario para proporcionar los resúmenes de mensajes o firmas digitales para los mensajes.

Integridad de datos

Implementación de la integridad de datos en los mensajes

Cuando se utiliza TLS, la opción de CipherSpec determina el nivel de integridad de datos en la empresa. Si utiliza IBM MQ Advanced Message Service (AMS), puede especificar la integridad de un mensaje exclusivo.

Implementación de la integridad de datos en salidas de mensajes

Un mensaje se puede firmar digitalmente mediante una salida de mensajes en el extremo emisor de un canal. Luego se puede comprobar la firma digital mediante una salida de mensajes en el extremo receptor de un canal para detectar si se ha modificado deliberadamente.

Se puede proporcionar cierta protección utilizando un resumen de mensaje en lugar de una firma digital. Un resumen de mensaje puede resultar eficaz frente a una manipulación casual o indiscriminada, pero no evita que una persona más informada modifique o sustituya el mensaje y genere para el mismo un resumen completamente nuevo. Esto resulta especialmente cierto si el algoritmo utilizado para generar el resumen de mensaje es muy conocido.

Implementación de la integridad de datos en salidas de emisión y recepción

En un canal de mensajes, las salidas de mensajes resultan más adecuadas para proporcionar este servicio porque una salida de mensajes tiene acceso al mensaje completo. En un canal MQI, los parámetros de llamadas MQI pueden contener datos de aplicación que se tengan que proteger, y sólo las salidas de emisión y recepción pueden proporcionar esta protección.

Implementación de la integridad de los datos en la salida de API o la salida cruzada de API

Una salida de API o salida cruzada de API puede firmar digitalmente un mensaje cuando la aplicación emisora transfiere el mensaje. La firma digital puede comprobarse mediante una segunda salida cuando la aplicación receptor recupera el mensaje para detectar si el mensaje ha sido modificado de forma deliberada.

Se puede proporcionar cierta protección utilizando un resumen de mensaje en lugar de una firma digital. Un resumen de mensaje puede resultar eficaz frente a una manipulación casual o indiscriminada, pero no evita que una persona más informada modifique o sustituya el mensaje y genere para el mismo un resumen completamente nuevo. Esto resulta especialmente cierto si el algoritmo utilizado para generar el resumen de mensaje es muy conocido.

Información adicional

Consulte la sección “Habilitación de CipherSpecs” en la página 426 para obtener más información sobre cómo garantizar la integridad de los datos.

Tareas relacionadas

Conexión de dos gestores de colas utilizando TLS

Conexión de un cliente a un gestor de colas de forma segura

Auditoría

Puede comprobar las intrusiones de seguridad, o intentos de intrusión, mediante mensajes de sucesos. También puede comprobar la seguridad del sistema utilizando IBM MQ Explorer.

Para detectar intentos de realizar acciones no autorizadas a tales como conectarse a un gestor de colas o transferir un mensaje a una cola, examine los mensajes de suceso generados por los gestores de colas, particularmente los mensajes de sucesos de autorización. Si desea más información sobre los mensajes de suceso del gestor de colas, consulte Sucesos del gestor de colas y si desea más información sobre la supervisión de sucesos en general, consulte Supervisión de sucesos.

Mantenimiento de la seguridad de los clústeres

Autorice o impida que los gestores de colas unan clústeres o coloquen mensajes en colas de clúster. Obligue a un gestor de colas a abandonar un clúster. Tenga en cuenta algunas consideraciones adicionales al configurar TLS para los clústeres.

Impedir que los gestores de colas no autorizados envíen mensajes

Impida que los gestores de colas no autorizados envíen mensajes a su gestor de colas utilizando una salida de seguridad de canal.

Antes de empezar

La agrupación en clúster no tiene ningún efecto en la manera en que funcionan las salidas de seguridad. Puede restringir el acceso a un gestor de colas igual que lo haría en un entorno de gestión de colas distribuidas.

Acerca de esta tarea

Impida que gestores de colas seleccionados envíen mensajes a su gestor de colas:

Procedimiento

1. Defina un programa de salida de seguridad de canal en la definición de canal CLUSRCVR.
2. Escriba un programa que autentique a los gestores de colas que intentan enviar mensajes en su canal de clúster receptor y que les deniegue el acceso si no están autorizados.

Qué hacer a continuación

Los programas de salida de seguridad de canal se invocan en la iniciación y la terminación del MCA.

Cómo hacer que los gestores de colas sin autorización pongan mensajes en sus colas

Utilice el atributo de canal Autorización de transferencia en el canal de clúster receptor para impedir que los gestores de colas no autorizados transfieran mensajes a sus colas. Autorice un gestor de colas remoto comprobando el ID de usuario en el mensaje utilizando RACF en z/OS, o el OAM en otras plataformas.

Acerca de esta tarea

Utilice los recursos de seguridad de una plataforma y el mecanismo de control de acceso de IBM MQ para controlar el acceso a las colas.

Procedimiento

1. Para impedir que ciertos gestores de colas transfieran mensajes a una cola, utilice los recursos de seguridad disponibles en su plataforma.

Por ejemplo:

- RACF u otros gestores de seguridad externos en IBM MQ for z/OS
- El gestor de autorizaciones sobre objetos (OAM) en otras plataformas.

2. Utilice el atributo de autorización de transferencia, PUTAUT, en la definición de canal CLUSRCVR.

   El atributo PUTAUT le permite especificar qué identificadores de usuario se van a utilizar para establecer la autorización para transferir un mensaje a una cola.

   Las opciones del atributo PUTAUT son:

   DEF
   Utilice el ID de usuario predeterminado. En z/OS, la comprobación puede implicar el uso tanto del ID de usuario recibido de la red como del derivado de MCAUSER.

   CTX
   Utilizar el ID de usuario en la información de contexto asociada al mensaje. En z/OS, la comprobación puede implicar el uso del ID de usuario recibido de la red, del derivado de MCAUSER, o de ambos. Utilice esta opción si el enlace es fiable y está autenticado.

   ONLYMCA (solamente z/OS)
   Como en DEF, pero no se utiliza ningún ID de usuario recibido de la red. Utilice esta opción si el enlace no es fiable. Permita en él sólo un conjunto específico de acciones, que se definen para MCAUSER.

   ALTMCA (solamente z/OS)
   Como en CTX, pero no se utiliza ningún ID de usuario recibido de la red.

Autorización de transferencia de mensajes a colas de clústeres remotos

En z/OS configure la autorización para transferir a una cola de clúster utilizando RACF. En otras plataformas, autorice el acceso para conectarse a los gestores de colas y para transferir las colas en dichos gestores de colas.

Acerca de esta tarea

El comportamiento predeterminado es realizar el control de acceso para SYSTEM.CLUSTER.TRANSMIT.QUEUE. Tenga en cuenta que este comportamiento se aplica, incluso si está utilizando varias colas de transmisión.

El comportamiento descrito en este tema solamente se aplica si ha configurado el atributo **ClusterQueueAccessControl** en el archivo qm.ini para que sea *RQMName*, tal como se describe en el tema Stanza de seguridad y si ha reiniciado el gestor de colas.

Procedimiento

- Para z/OS, emita los mandatos siguientes:

```
RDEFINE MQQUEUE QMgrName.QUEUE. QueueName UACC(NONE)
PERMIT QMgrName.QUEUE. QueueName CLASS(MQADMIN) ID(GroupName) ACCESS(UPDATE)
```

- Para sistemas UNIX, Linux, and Windows, emita los mandatos siguientes:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

- Para IBM i, emita los mandatos siguientes:

```
GRTMQMAUT OBJ(' QMgrName ') OBJTYPE(*MQM) USER(GroupName) AUT(*CONNECT)
GRTMQMAUT OBJ(' QueueName ') OBJTYPE(*Q) USER(GroupName) AUT(*PUT) MQMNAME(' QMgrName ')
```

El usuario sólo puede transferir mensajes a la cola de clúster especificada, y no a otras colas de clúster.

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

GroupName
Nombre del grupo al que se va a otorgar el acceso.

QueueName
Nombre de la cola o perfil genérico para el que se van a cambiar autorizaciones.

Qué hacer a continuación

Si especifica una cola de respuesta cuando transfiere un mensaje a una cola de clúster, la aplicación de consumo debe tener autorización para enviar la respuesta. Establezca esta autorización siguiendo las instrucciones de "Otorgar autorización para transferir mensajes a una cola de clúster remota" en la página 403.

Conceptos relacionados
Stanza de seguridad en qm.ini

Impedir que gestores de colas se unan a un clúster

Si un gestor de colas falso se une a un clúster, es difícil impedir que reciba mensajes que usted no desea que reciba.

Procedimiento

Si desea asegurarse de que sólo determinados gestores de colas autorizados se unen a un clúster, puede elegir entre tres técnicas:

- Mediante el uso de registros de autenticación de canal, puede bloquear la conexión de canal de clúster basándose en: la dirección IP remota, el nombre del gestor de colas remoto o el Nombre distinguido TLS proporcionado por el sistema remoto.
- Escribir un programa de salida para impedir que los gestores de colas no autorizados graben en la cola `SYSTEM.CLUSTER.COMMAND.QUEUE`. No restrinja el acceso a `SYSTEM.CLUSTER.COMMAND.QUEUE` de manera que ningún gestor de colas pueda grabar en ella, o impedirá que cualquier gestor de colas que se una al clúster.
- Un programa de salida de seguridad en la definición de canal `CLUSRCVR`.

Salidas de seguridad en canales de clúster

Consideraciones adicionales al utilizar salidas de seguridad en canales de clúster.

Acerca de esta tarea

Cuando un canal de clúster emisor se inicia por primera vez, utiliza atributos definidos manualmente por un administrador del sistema. Cuando el canal se detiene y se reinicia, toma los atributos de la definición de canal de clúster emisor correspondiente. La definición de canal de clúster emisor original se sobrescribe con los nuevos atributos, incluido el atributo `SecurityExit`.

Procedimiento

1. Debe definir una salida de seguridad tanto en el extremo del clúster emisor como en el extremo del clúster receptor de un canal.

   La conexión inicial debe establecerse con un reconocimiento de salida de seguridad, aunque el nombre de salida de seguridad se envíe desde la definición de clúster receptor.

2. Valide el PartnerName en la estructura MQCXP de la salida de seguridad.

   La salida debe permitir que el canal se inicie únicamente si el gestor de colas asociado está autorizado.
3. Diseñe la salida de seguridad de la definición de clúster receptor para que se inicie con el receptor.
4. Si la diseña como iniciada con el emisor, un gestor de colas no autorizado sin una salida de seguridad puede unirse al clúster porque no se realiza ninguna comprobación de seguridad.

   Hasta que el canal no se haya detenido y reiniciado, no se podrá enviar el nombre SCYEXIT desde la definición de clúster receptor ni se podrán realizar comprobaciones de seguridad completas.
5. Para ver la definición de canal de clúster emisor que se está utilizando en este momento, utilice el mandato:

   ```
   DISPLAY CLUSQMGR( queue manager ) ALL
   ```

   El mandato muestra los atributos que se han enviado desde la definición de clúster receptor.
6. Para ver la definición original, utilice el mandato:

   ```
   DISPLAY CHANNEL( channel name ) ALL
   ```
7. Es posible que tenga que definir una salida de definición automática de canal, CHADEXIT, en el gestor de colas del clúster emisor, si los gestores de colas se encuentran en plataformas diferentes.

   Utilice la salida de definición automática de canal para establecer el atributo SecurityExit en un formato adecuado para la plataforma de destino.
8. Despliegue y configure la salida de seguridad.

   z/OS **z/OS**

   El módulo de carga de salida de seguridad debe estar en el conjunto de datos especificado en la sentencia CSQXLIB DD del procedimiento de espacio de direcciones del iniciador de canal.

   ULW **Sistemas Windows, UNIX and Linux**

   - La biblioteca de enlace dinámico de salida de seguridad debe estar en la vía de acceso especificada en el atributo SCYEXIT de la definición de canal.
   - La biblioteca de enlace dinámico de salida de definición automática de canal debe estar en la vía de acceso especificada en el atributo CHADEXIT de la definición de gestor de colas.

Forzar que los gestores de colas no deseados abandonen un clúster

Puede forzar que un gestor de colas no deseado abandone un clúster emitiendo el mandato RESET CLUSTER en un gestor de colas de repositorio completo.

Acerca de esta tarea

Puede forzar a que un gestor de colas no deseado deje un clúster. Por ejemplo, si se suprime un gestor de colas pero sus canales de clúster receptor siguen estando definidos en el clúster, es posible que desee una reorganización.

Sólo los gestores de colas de repositorio completo tienen autorización para expulsar a un gestor de colas de un clúster.

Nota: Aunque la utilización del mandato RESET CLUSTER fuerza la eliminación de un gestor de colas de un clúster, si utiliza RESET CLUSTER por sí solo no impide que el gestor de colas se reincorpore al clúster más adelante. Para asegurarse de que el gestor de colas no se vuelva a unir al clúster, siga los pasos detallados en “Impedir que gestores de colas se unan a un clúster” en la página 467.

Siga este procedimiento para expulsar al gestor de colas OSLO del clúster NORWAY:

Procedimiento

1. En un gestor de colas de depósito completo, emita el mandato:

```
RESET CLUSTER(NORWAY) QMNAME(OSLO) ACTION(FORCEREMOVE)
```

2. O utilice el MQID en lugar de QMNAME en el mandato:

```
RESET CLUSTER(NORWAY) QMID(qmid) ACTION(FORCEREMOVE)
```

   Nota: QMID es una serie, por lo que el valor de qmid debe estar entre comillas simples, por ejemplo, QMID('FR01_2019-07-15_14.42.42').

Resultados

El gestor de colas que se elimina forzadamente no cambia; sus definiciones de clúster locales muestran que está en el clúster. Las definiciones en todos los demás gestores de colas no muestran que está en el clúster.

Cómo impedir que los gestores de colas reciban mensajes

Puede impedir que un gestor de colas reciba mensajes si no está autorizado para recibirlos utilizando programas de salida.

Acerca de esta tarea

Es difícil impedir a un gestor de colas de un clúster que defina una cola. Existe el peligro de que un gestor de colas falso pueda unirse a un clúster y defina su propia instancia de una de las colas en el clúster. Ahora puede recibir mensajes que no está autorizado a recibir. Para impedir que un gestor de colas reciba mensajes, utilice una de las opciones siguientes indicadas en el procedimiento.

Procedimiento

- Un programa de salida de canal en cada canal de clúster emisor. El programa de salida utiliza el nombre de conexión para determinar la adecuación del gestor de colas de destino al que se deban enviar los mensajes.
- Un programa de salida de carga de trabajo del clúster, que utiliza los registros de destino para determinar la adecuación de la cola de destino y el gestor de colas al que se deban enviar los mensajes.

SSL/TLS y clústeres

Al configurar TLS para clústeres, tenga en cuenta que se propaga una definición de canal CLUSRCVR a otros gestores de colas como un canal CLUSSDR definido automáticamente. Si un canal CLUSRCVR utiliza TLS, debe configurar TLS en todos los gestores de colas que se comuniquen utilizando el canal.

Para obtener más información sobre TLS, consulte “Protocolos de seguridad TLS en IBM MQ” en la página 24. Los consejos que se ofrecen en dicho tema generalmente son aplicables a los canales del clúster, pero tal vez desee considerar lo siguiente:

En un clúster de IBM MQ se propaga frecuentemente una definición de canal CLUSRCVR específica a muchos otros gestores de colas, donde se transforma en un CLUSSDR definido automáticamente. Posteriormente, el CLUSSDR definido automáticamente se utiliza para iniciar un canal para el CLUSRCVR. Si el CLUSRCVR está configurado para la conectividad TLS, se aplican las siguientes consideraciones:

- Todos los gestores de colas que deseen comunicarse con este CLUSRCVR debe tener acceso al soporte de TLS. Esta provisión de TLS debe dar soporte a la CipherSpec para el canal.
- Los diferentes gestores de colas a los que se han propagado los canales de clúster emisor definidos automáticamente tendrán cada uno un nombre distinguido diferente asociado. Si se va a utilizar la

comprobación de nombres distinguidos de iguales en el CLUSRCVR, éste debe configurare de manera que todos los nombres distinguidos que puedan recibirse se comparen correctamente.

Por ejemplo, supongamos que todos los gestores de colas que alojarán canales de clúster emisor que conectarán a un CLUSRCVR determinado, tienen certificados asociados. Supongamos también que los nombres distinguidos en todos estos certificados definen el país como UK, la organización como IBM, la unidad organizativa como IBM MQ Development, y que todos tienen nombres comunes en el formato DEVT.QMnnn, donde nnn es un valor numérico.

En este caso, un valor de SSLPEER de `C=UK, O=IBM, OU=IBM MQ Development, CN=DEVT.QM*` en el CLUSRCVR permitirá que todos los canales de emisor de clúster se conecten correctamente, pero impedirá que se conecten canales de emisor de clúster no deseados.

- Si se utilizan series CipherSpec personalizadas, tenga en cuenta que los formatos de serie personalizados no están permitidos en todas las plataformas. Un ejemplo de ello es que la serie CipherSpec RC4_SHA_US tiene un valor de 05 en IBM i , pero no es una especificación válida en sistemas UNIX, Linux o Windows. Por lo tanto, si se utilizan parámetros SSLCIPH personalizados en un CLUSRCVR, todos los canales de clúster emisor definidos automáticamente resultantes deben residir en plataformas en las que el soporte TLS subyacente implemente esta CipherSpec y en las que se pueda especificar con el valor personalizado. Si no puede seleccionar un valor para el parámetro SSLCIPH que se pueda entender en todo el clúster, necesitará una salida de definición automática de canal para transformarla en algo que puedan interpretar las plataformas que se utilizan. Utilice las series CipherSpec de texto cuando sea posible (por ejemplo TLS_RSA_WITH_AES_128_CBC_SHA).

Un parámetro SSLCRLNL se aplica a un gestor de colas individual y no se propaga a otros gestores de colas de un clúster.

Actualización de gestores de colas y canales en clúster a SSL/TLS

Actualice los canales de clúster de uno en uno, cambiando todos los canales CLUSRCVR antes que los canales CLUSSDR.

Antes de empezar

Tenga en cuenta las consideraciones siguientes, ya que estas podrían afectar a la elección de CipherSpec para un clúster:

- Algunas CipherSpecs no están disponibles en todas las plataformas. Procure elegir una CipherSpec que esté soportada por todos los gestores de colas en el clúster.
- Algunas CipherSpecs podrían ser nuevas en el release de IBM MQ actual y no se soportan en releases anteriores. Un clúster que contiene gestores de colas que se ejecutan en releases MQ diferentes sólo podrá utilizar las CipherSpecs soportadas por cada release.

  Para utilizar una nueva CipherSpec dentro de un clúster, primero debe migrar todos los gestores de colas de clúster al release actual.
- Algunas CipherSpecs requieren el uso de un tipo específico de certificado digital, especialmente aquellas que utilizan cifrado Elliptic Curve.

Atención: No es posible utilizar una combinación de certificados firmados por Elliptic Curve y los certificados firmados por RSA en los gestores de colas que desea unir como parte de un clúster.

Los gestores de colas de un clúster deben utilizar todos los certificados firmados por RSA, o bien utilizar todos los certificados firmados por EC, no una combinación de ambos.

Consulte “Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45 para obtener más información.

Actualice todos los gestores de colas del clúster a IBM MQ V8 o superior, si todavía no están en estos niveles. Distribuya los certificados y las claves para que TLS funcione desde cada uno de ellos.

Si desea actualizar tom o utilizar ANY_TLS12 CipherSpecs, debe actualizar todos los gestores de colas del clúster a IBM MQ 9.1.2 o superior.

Si desea actualizar o utilizar cualquiera de los otros alias CipherSpecs (ANY_TLS13, ANY_TLS12, ANY_TLS12_OR_HIGHER, etc.), debe actualizar todos los gestores de colas del clúster a IBM MQ 9.1.4 o superior.

Acerca de esta tarea

Cambie los canales CLUSRCVR antes que los canales CLUSSDR.

Procedimiento

1. Conmute los canales CLUSRCVR a TLS en cualquier orden que desee, cambiando los canales CLUSRCVR de uno en uno, y permita que los cambios circulen por el clúster antes de cambiar el siguiente.

   Importante: Asegúrese de no cambiar la ruta inversa hasta que los cambios para el canal actual se hayan distribuido por el clúster.
2. Opcional: Cambie todos los canales CLUSSDR manuales a TLS.

   Esto no tiene ningún efecto en el funcionamiento del clúster, a menos que se utilice el mandato REFRESH CLUSTER con la opción REPOS(YES).

   Nota: Para clústeres de gran tamaño, el uso del mandato **REFRESH CLUSTER** puede interrumpir el clúster mientras está en curso, y de nuevo a intervalos de 27 días a partir de entonces cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.
3. Utilice el mandato DISPLAY CLUSQMGR para asegurarse de que la nueva configuración de seguridad se ha propagado en todo el clúster.
4. Reinicie los canales para que utilicen TLS y ejecute REFRESH SECURITY (SSL).

Conceptos relacionados

“Habilitación de CipherSpecs” en la página 426
Habilite una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC **DEFINE CHANNEL** o el mandato MQSC **ALTER CHANNEL**.

“Certificados digitales y compatibilidad de CipherSpec en IBM MQ” en la página 45
En este tema se proporciona información sobre cómo elegir las CipherSpecs y los certificados digitales adecuados para su política de seguridad, describiendo la relación entre las CipherSpecs y los certificados digitales en IBM MQ.

Información relacionada

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

Inhabilitación de SSL/TLS en gestores de colas y canales en clúster

Para desactivar TLS, establezca el parámetro SSLCIPH en ' '. Inhabilite TLS en los canales del clúster de individual, cambiando todos los canales de clúster receptores antes que los canales de clúster emisores.

Acerca de esta tarea

Cambie un canal de clúster emisor cada vez y permita que los cambios fluyan por el clúster antes de cambiar el siguiente.

Importante: Asegúrese de no cambiar la ruta inversa hasta que los cambios para el canal actual se hayan distribuido por el clúster.

Procedimiento

1. Defina el valor del parámetro SSLCIPH en ' ', una serie vacía entre comillas sencillas IBM i, o *NONE en IBM i .

Puede desactivar TLS en los canales de clúster receptores en el orden que desee.

Tenga en cuenta que los cambios fluyen en dirección opuesta por canales en los que se deja TLS activo.

2. Compruebe que el nuevo valor se refleja en todos los demás gestores de colas utilizando el mandato **DISPLAY CLUSQMGR(*)** ALL.
3. Desactive TLS en todos los canales de clúster emisores manuales.

   Esto no tiene ningún efecto en el funcionamiento del clúster, a menos que se utilice el mandato **REFRESH CLUSTER** con la opción REPOS(YES).

   Para los clústeres de gran tamaño, utilice el mandato **REFRESH CLUSTER** puede generar problemas a intervalos regulares posteriormente, cuando los objetos de clúster envían automáticamente actualizaciones de estado a todos los gestores de colas interesados. Consulte La actualización en un clúster de gran tamaño puede afectar el rendimiento y la disponibilidad del clúster para obtener más información.
4. Detenga y reinicie los canales de clúster emisores.

Seguridad de publicación/suscripción

Los componentes e interacciones que están implicados en la publicación/suscripción se describen como una introducción a las explicaciones más detalladas y los ejemplos que siguen.

Hay una serie de componentes implicados en la publicación y suscripción a un tema. Algunas de las relaciones de seguridad entre ellos se ilustran en la Figura 22 en la página 473 y se describen en el siguiente ejemplo.

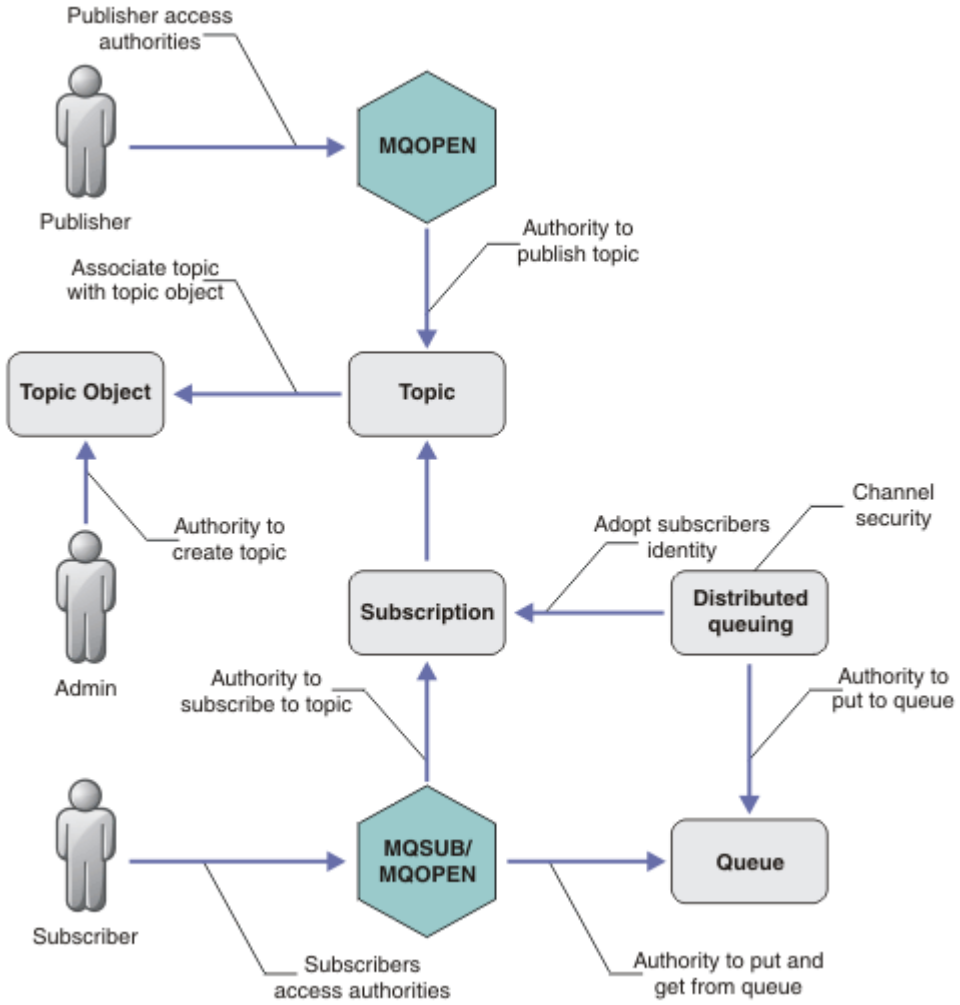


Figura 22. Relaciones de seguridad de publicación/suscripción entre diversos componentes

Temas

Los temas se identifican mediante series de tema, y normalmente se organizan en árboles; consulte Árboles de temas. Debe asociar un tema con un objeto de tema para controlar el acceso al tema. En la sección "Modelo de seguridad de temas" en la página 475 se describe cómo proteger los temas mediante objetos de tema.

Objetos de temas administrativos

Puede controlar quién tiene acceso a un tema, y con qué finalidad, mediante el mandato **setmqaut** con una lista de objetos de temas administrativos. Consulte los ejemplos en "Otorgar acceso a un usuario para suscribirse a un tema" en la página 480 y en "Otorgar acceso a un usuario para publicar en un tema" en la página 487. z/OS Para controlar el acceso a los objetos de tema en z/OS, consulte Perfiles la seguridad de temas.

Suscripciones

Suscríbase a uno o varios temas creando una suscripción mediante una serie de tema, que puede incluir comodines, para que coincida con la serie de tema de las publicaciones. Para obtener más detalles, consulte:

Suscripción mediante un objeto de tema

"Suscripción utilizando el nombre de objeto de tema" en la página 476

Suscripción mediante un tema

"Suscripción utilizando una serie del tema donde el nodo de tema no existe" en la página 477

Suscripción mediante un tema con comodines

"Suscripción utilizando una serie de tema que contiene caracteres comodín" en la página 477

Una suscripción contiene información sobre la identidad del suscriptor y la identidad de la cola de destino en la que se van a colocar las publicaciones. También contiene información sobre cómo debe colocarse la publicación en la cola de destino.

Del mismo modo que puede definir qué suscriptores tienen autorización para suscribirse a determinados temas, puede restringir las suscripciones para que sean utilizadas por un suscriptor individual. También puede controlar qué información sobre el suscriptor utiliza el gestor de colas cuando las publicaciones se colocan en la cola de destino. Consulte "Seguridad de suscripción" en la página 493.

Colas

La cola de destino es una cola importante que debe protegerse. Es local para el suscriptor, y las publicaciones que coinciden con la suscripción se colocan en ella. Debe tener en cuenta el acceso a la cola de destino desde dos perspectivas:

1. Transferencia de una publicación a la cola de destino.
2. Obtención de la publicación de la cola de destino.

El gestor de colas transfiere una publicación a la cola de destino utilizando una identidad proporcionada por el suscriptor. El suscriptor, o un programa al que ha sido delegado la tarea de obtener publicaciones, toma mensajes de la cola. Consulte "Autorización para colas de destino" en la página 478.

No hay alias de objeto de tema, pero puede utilizar una cola de alias como alias para un objeto de tema. Si lo hace, y se comprueba la autorización para utilizar el tema de publicación o suscripción, el gestor de colas comprueba la autorización para utilizar la cola.

"Seguridad de publicación/suscripción entre gestores de colas" en la página 494

Su permiso para publicar o suscribirse a un tema se comprueba en el gestor de colas local utilizando identidades y autorizaciones locales. La autorización no depende de si el tema se define o no, ni de dónde está definido. Por consiguiente, debe realizar la autorización de temas en cada gestor de colas de un clúster cuando se utilizan temas en clúster.

Nota: El modelo de seguridad para los temas difiere del modelo de seguridad para las colas. Puede conseguir el mismo resultado para las colas mediante la definición, a nivel local, de un alias de cola para cada cola en clúster.

Los gestores de colas intercambian suscripciones en un clúster. En la mayoría de configuraciones de clúster de IBM MQ, los canales se configuran con PUTAUT=DEF para colocar mensajes en las colas de destino usando la autorización del proceso del canal. Se puede modificar la configuración del canal para utilizar PUTAUT=CTX a fin de exigir que el usuario suscriptor tenga autorización para propagar una suscripción a otro gestor de colas en un clúster.

En "Seguridad de publicación/suscripción entre gestores de colas" en la página 494 se describe cómo cambiar las definiciones de canal para controlar quién tiene permiso para propagar suscripciones en otros servidores del clúster.

Autorización

Puede aplicar autorización a objetos de tema, como ocurre con las colas y otros objetos. Hay tres operaciones de autorización, `pub`, `sub` y `resume` que pueden aplicarse sólo a temas. Los detalles se describen en Especificación de autorizaciones para tipos de objeto diferentes.

Llamadas de función

En programas de publicación y suscripción, como en programas de transmisión a colas, las comprobaciones de autorización se realizan cuando se abren, crean, cambian o eliminan objetos. No se realizan comprobaciones cuando se llevan a cabo llamadas MQPUT o MQGET de MQI para transferir y obtener publicaciones.

Para publicar un tema, realice una llamada MQOPEN en el tema, que realiza las comprobaciones de autorización. Publique mensajes en el descriptor de tema mediante el mandato MQPUT, que no realiza comprobaciones de autorización.

Para suscribirse a un tema, generalmente debe ejecutar un mandato MQSUB para crear o reanudar una suscripción, y también abrir la cola de destino para que pueda recibir publicaciones. De forma alternativa, ejecute un mandato MQOPEN por separado para abrir la cola de destino y, a continuación, ejecute un mandato MQSUB para crear o reanudar la suscripción.

Independientemente de las llamadas que utilice, el gestor de colas comprueba que puede suscribirse al tema y obtener las publicaciones resultantes de la cola de destino. Si la cola de destino no está gestionada, también se realizan comprobaciones de la autorización para ver si el gestor de colas puede transferir publicaciones a la cola de destino. Utiliza la identidad que adoptó a partir en una suscripción coincidente. Se supone que el gestor de colas siempre es capaz de colocar las publicaciones en las colas de destino gestionado.

Roles

Los usuarios están involucrados en cuatro roles al ejecutar aplicaciones de publicación/suscripción:

1. Publicador
2. Suscriptor
3. Administrador de temas
4. Administrador de IBM MQ, miembro del grupo mqm

Defina grupos con las autorizaciones apropiadas que correspondan a los roles de publicación, suscripción y administración de temas. A continuación, puede asignar principales a estos grupos autorizándoles a realizar tareas específicas de publicación y suscripción.

Además, debe ampliar las autorizaciones de operaciones administrativas para el administrador de colas y canales responsable de mover publicaciones y suscripciones.

Modelo de seguridad de temas

Los objetos de tema definidos son los únicos que pueden tener atributos de seguridad asociados. Para obtener una descripción de los objetos de tema, consulte Objetos de tema administrativo. Los atributos de seguridad especifican si un ID de usuario determinado, o un grupo de seguridad, pueden realizar una operación de suscripción o publicación en cada objeto de tema.

Los atributos de seguridad están asociados con el nodo de administración adecuado en el árbol de temas. Cuando se efectúa una comprobación de autorización para un ID de usuario determinado durante una operación de suscripción o publicación, la autorización otorgada se basa en los atributos de seguridad del nodo del árbol de temas asociado.

Los atributos de seguridad son una lista de control de acceso que indica qué autorización tiene un ID de usuario o grupo de seguridad determinado del sistema operativo sobre el objeto de tema.

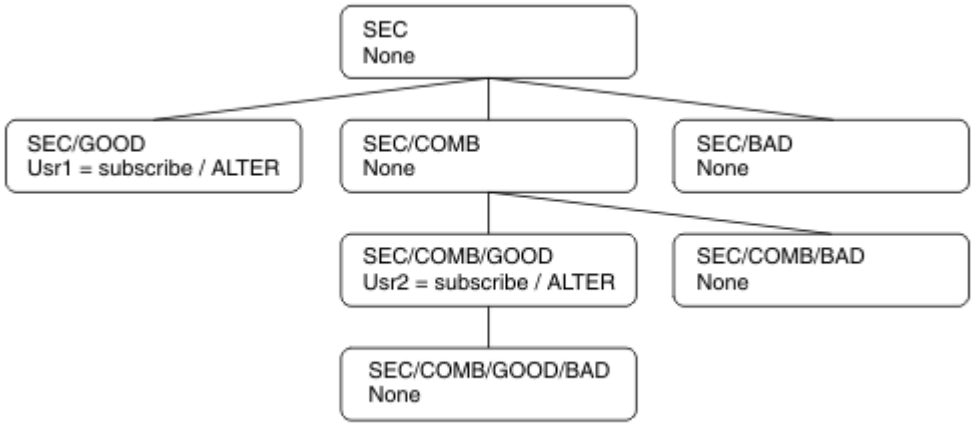
Considere el ejemplo siguiente donde los objetos de tema se han definido con atributos de seguridad o autorizaciones:

Tabla 80. Ejemplo de autorizaciones de objetos de tema

Nombre de tema	Serie de tema	Autorizaciones, no z/OS	Autorizaciones de z/OS
SECROOT	SEC	Ninguna	Ninguna
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD
SECBAD	SEC/BAD	Ninguna	Ninguna HLQ.SUBSCRIBE.SECBAD

Tabla 80. Ejemplo de autorizaciones de objetos de tema (continuación)			
Nombre de tema	**Serie de tema**	**Autorizaciones, no z/OS**	**Autorizaciones de z/OS**
SECCOMB	SEC/COMB	Ninguna	Ninguna HLQ.SUBSCRIBE.SECCOMB
SECCOMBB	SEC/COMB/ GOOD/BAD	Ninguna	Ninguna HLQ.SUBSCRIBE.SECCOMBB
SECCOMBG	SEC/COMB/GOOD	usr2+subscribe	ALTER HLQ.SUBSCRIBE.SECCOMBG
SECCOMBN	SEC/COMB/BAD	Ninguna	Ninguna HLQ.SUBSCRIBE.SECCOMBN

El árbol de temas con los atributos de seguridad asociados en cada nodo puede representar del siguiente modo:



Los ejemplos enumerados otorgan las autorizaciones siguientes:

- En el nodo raíz del árbol de /SEC, ningún usuario tiene autorización en dicho nodo.
- A usr1 se le ha otorgado autorización de suscripción para el objeto /SEC/GOOD
- A usr2 se le ha otorgado autorización de suscripción para el objeto /SEC/COMB/GOOD

Suscripción utilizando el nombre de objeto de tema

Al suscribirse a un objeto de tema especificando el nombre MQCHAR48, se localiza el nodo correspondiente del árbol de temas. Si los atributos seguridad asociados con el nodo indican que el usuario tiene autorización para suscribirse, se otorga acceso.

Si no se otorga acceso al usuario, el nodo padre del árbol determina si el usuario tiene autorización para suscribirse en el nivel de nodo padre. Si es así, se otorga acceso. En caso contrario, se considera el padre de dicho nodo. La recurrencia continúa hasta que se encuentra un nodo que otorga autorización de suscripción al usuario. La recurrencia se detiene cuando se considera el nodo raíz sin haber sido otorgado autorización. En este último caso, se deniega el acceso.

En pocas palabras, si cualquier nodo en la vía otorga al usuario o aplicación autorización para suscribirse, el suscriptor está autorizado para suscribirse a dicho nodo, o a cualquier nodo por debajo de dicho nodo en el árbol de temas.

El nodo raíz en el ejemplo es SEC.

Se otorga autorización de suscripción al usuario si la lista de control de acceso indica que el propio ID de usuario tiene autorización, o que un grupo de seguridad del sistema operativo del que el ID de usuario es miembro tiene autorización.

Así, por ejemplo:

- Si `usr1` intenta suscribirse mediante una serie de tema de `SEC/GOOD`, la suscripción se permitirá porque el ID de usuario tiene acceso al nodo asociado con dicho tema. Sin embargo, si `usr1` ha intentado suscribirse utilizando la serie de tema `SEC/COMB/GOOD`, la suscripción no se permitirá porque el ID de usuario no tiene acceso a la nodo asociado a él.
- Si `usr2` intenta suscribirse mediante una serie de tema de `SEC/COMB/GOOD`, la suscripción se permitirá porque el ID de usuario tiene acceso al nodo asociado con el tema. Sin embargo, si `usr2` ha intentado suscribirse a `SEC/GOOD`, la suscripción no se permitirá porque el ID de usuario no tiene acceso al nodo asociado a él.
- Si `usr2` intenta suscribirse utilizando una serie de tema de `SEC/COMB/GOOD/BAD`, la suscripción se permitirá porque el ID de usuario tiene acceso al nodo padre `SEC/COMB/GOOD`.
- Si `usr1` o `usr2` intentan suscribirse utilizando una serie de tema de `/SEC/COMB/BAD`, no se permitirá porque no tienen acceso al nodo de tema asociado o a los nodos padre de dicho tema.

Una operación de suscripción que especifique el nombre de un objeto de tema que no existe dará lugar a un error MQRC_UNKNOWN_OBJECT_NAME.

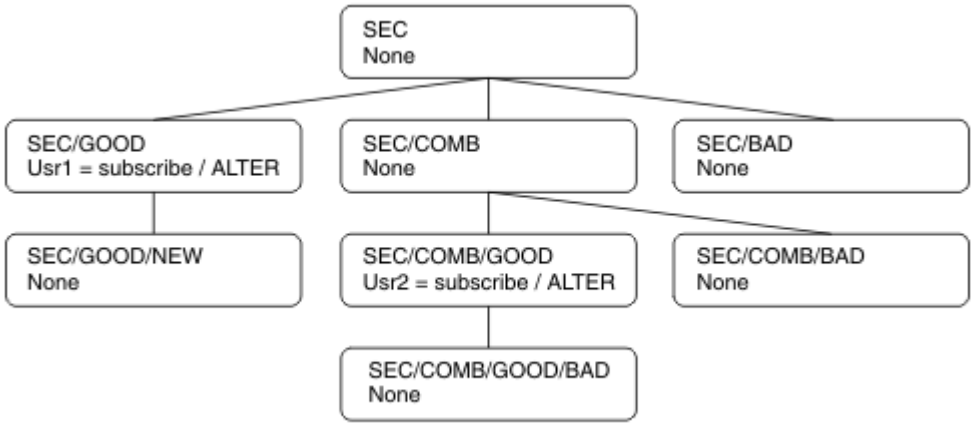
Suscripción utilizando una serie del tema donde el nodo de tema existe

El comportamiento es el mismo que cuando se especifica el tema por el nombre del objeto MQCHAR48.

Suscripción utilizando una serie del tema donde el nodo de tema no existe

Considere el caso de una aplicación de suscripción que especifica una serie de tema que representa un nodo de tema que no existe actualmente en el árbol de temas. La comprobación de autorización se realiza como se describe en el apartado anterior. La selección empieza con el nodo padre representado por la serie de tema. Si se otorga la autorización, se crea un nodo nuevo que representa la serie de tema en el árbol de temas.

Por ejemplo, `usr1` intenta suscribirse a un tema `SEC/GOOD/NEW`. La autorización se otorga porque `usr1` tiene acceso al nodo padre `SEC/GOOD`. Se crea un nodo de tema nuevo en el árbol como se muestra en el siguiente diagrama. El nodo de tema nuevo no es un objeto de tema y no tiene ningún atributos de seguridad asociado directamente; los atributos los hereda de su padre.



Suscripción utilizando una serie de tema que contiene caracteres comodín

Considere el caso de una suscripción mediante una serie de tema que contiene un carácter comodín. La comprobación de autorización se efectúa en el nodo del árbol de temas que coincide con la parte completa de la serie de tema.

Por lo tanto, si una aplicación se suscribe a SEC/COMB/GOOD/*, se lleva a cabo una comprobación de autorización como se describe en las dos secciones anteriores en el nodo SEC/COMB/GOOD del árbol de temas.

Del mismo modo, si una aplicación debe suscribirse a SEC/COMB/*/GOOD, se lleva a cabo una comprobación de autorización en el nodo SEC/COMB.

Autorización para colas de destino

Al suscribirse a un tema, uno de los parámetros es el manejador hobj de una cola que se ha abierto para salida para recibir las publicaciones.

Si no se especifica hobj, pero está en blanco, se crea una cola gestionada si se cumplen las condiciones siguientes:

- Se ha especificado la opción MQSO_MANAGED.
- La suscripción no existe.
- Se ha especificado creación.

Si deja hobj en blanco, y modifica o reanuda una suscripción existente, la cola de destino indicada anteriormente debe ser gestionada o no gestionada.

La aplicación o el usuario que realiza la solicitud de MQSUB debe tener la autorización para transferir mensajes a la cola de destino especificada; en efecto, debe tener la autorización para transferir mensajes publicados a esa cola. La comprobación de autorización sigue las reglas existentes para la comprobación de la seguridad de las colas.

La comprobación de seguridad incluye el ID de usuario alternativo y las comprobaciones de seguridad de contexto si es necesario. Para poder establecer cualquiera de los campos de contexto de identidad, debe especificar la opción MQSO_SET_IDENTITY_CONTEXT, así como la opción MQSO_CREATE o MQSO_ALTER. No se puede establecer ninguno de los campos de contexto de identidad en una solicitud MQSO_RESUME.

Si el destino es una cola gestionada, no se realiza ninguna comprobación de seguridad en el destino gestionado. Si se le permite suscribirse a un tema, se supone que puede utilizar destinos gestionados.

Publicación utilizando el nombre de tema o una serie de tema donde el nodo de tema existe

El modelo de seguridad de la publicación es el mismo que el de la suscripción, excepto los caracteres comodín. Las publicaciones no contienen comodines; por lo tanto no hay ningún caso de una serie de tema que contenga caracteres comodín para tener en cuenta.

Las autorizaciones para publicar y suscribir son diferentes. Un usuario o grupo puede tener la autorización para llevar a cabo una de estas operaciones sin que necesariamente pueda realizar la otra.

Cuando se publica en un objeto de tema especificando el nombre MQCHAR48 o la serie del tema, se localiza el nodo correspondiente del árbol de temas. Si los atributos de seguridad asociados con el nodo de tema indican que el usuario tiene autorización para publicar, se otorga acceso.

Si no se ha otorgado el acceso, el nodo padre en el árbol determina si el usuario tiene autorización para publicar en dicho nivel. Si es así, se otorga acceso. Si no, la recurrencia continúa hasta que se encuentra un nodo que otorgue autorización de publicación para el usuario. La recurrencia se detiene cuando se considera el nodo raíz sin haber sido otorgado autorización. En este último caso, se deniega el acceso.

En pocas palabras, si algún nodo de la vía otorga a dicho usuario o aplicación autorización para publicar, el publicador puede publicar en dicho nodo o en cualquier lugar bajo dicho nodo en el árbol de temas.

Publicación utilizando el nombre de tema o una serie de tema donde el nodo de tema no existe

Como ocurre con la operación de suscripción, cuando una aplicación publica especificando una serie de tema que representa un nodo de tema que no existe actualmente en el árbol de temas, la comprobación

de autorización se realiza empezando por el padre del nodo representado por la serie de tema. Si se otorga la autorización, se crea un nodo nuevo que representa la serie de tema en el árbol de temas.

Publicación utilizando una cola de alias que se resuelve en un objeto de tema

Si publica utilizando una cola de alias que se resuelve en un objeto de tema, la comprobación de seguridad se produce tanto en la cola de alias como en el tema subyacente en el que se resuelve.

La comprobación de seguridad en la cola de alias verifica que el usuario tiene autorización para transferir mensajes a esa cola de alias y la comprobación de seguridad sobre el tema verifica que el usuario puede publicar en dicho tema. Cuando una cola alias se resuelve en otra cola, las comprobaciones no se realizan en la cola subyacente. La comprobación de autorización se realiza de forma distinta para temas y colas.

Cierre de una suscripción

Existe una comprobación de seguridad adicional si se cierra una suscripción utilizando la opción `MQCO_REMOVE_SUB` y si no se ha creado la suscripción bajo este manejador.

Se realiza una comprobación de seguridad para garantizar que tiene la autorización correcta para hacerlo, porque la acción da como resultado la eliminación de la suscripción. Si los atributos de seguridad asociados con el nodo de tema indican que el usuario tiene autorización, se otorga acceso. Si no es así, se considera el nodo padre del árbol para determinar si el usuario tiene autorización para cerrar la suscripción. La recurrencia continúa hasta que se otorga autorización o bien hasta que se alcanza el nodo raíz.

Definición, modificación y supresión de una suscripción

No se lleva a cabo ninguna comprobación de seguridad de la suscripción cuando se crea una suscripción administrativamente, en lugar de utilizar una solicitud de API `MQSUB`. El administrador ya ha recibido esta autorización a través del mandato.

Las comprobaciones de seguridad se realizan para garantizar que las publicaciones se pueden transferir a la cola de destino asociada con la suscripción. Las comprobaciones se realizan del mismo modo que para una solicitud `MQSUB`.

El ID de usuario que se utiliza para estas comprobaciones de seguridad depende del mandato que se emite. Si se especifica el parámetro **SUBUSER**, ello afecta al modo en que se lleva a cabo la comprobación, como se muestra en Tabla 81 en la página 479:

Tabla 81. ID de usuario utilizados para las comprobaciones de seguridad para mandatos

Mandato	SUBUSER especificado y en blanco	SUBUSER especificado y completo	SUBUSER no especificado
	Utilizar el ID de administrador		Utilizar el ID de usuario de la suscripción LIKE
	Utilizar el ID de administrador		Utilizar el ID.DEFAULT.SU de usuarioB - si está en de lablanco, suscripciónutilizar el ID SYSTEMde administrador

Tabla 81. ID de usuario utilizados para las comprobaciones de seguridad para mandatos (continuación)			
Mandato	**SUBUSER especificado y en blanco**	**SUBUSER especificado y completo**	**SUBUSER no especificado**
	Utilizar el ID de administrador		Utilizar el ID de usuario de la suscripción existente

La única comprobación de seguridad que se lleva a cabo al suprimir las suscripciones con el mandato DELETE SUB es la comprobación de seguridad de mandatos.

Ejemplo de configuración de seguridad de publicación/suscripción

En esta sección se describe un escenario que tiene configurado el control de accesos a los temas de forma que permite aplicar el control de seguridad según sea necesario.

Otorgar acceso a un usuario para suscribirse a un tema

Este tema es el primero de una lista de tareas que indica cómo otorgar acceso a los temas por más de un usuario.

Acerca de esta tarea

En esta tarea se presupone que no existen objetos de temas administrativos, ni se han definido los perfiles para la suscripción o publicación. Las aplicaciones crean nuevas suscripciones, en lugar de reanudar las existentes, y lo hacen utilizando sólo la serie de tema.

Una aplicación puede realizar una suscripción proporcionando un objeto de tema, o una serie de tema, o una combinación de ambos. Sea cual sea lo que seleccione la aplicación, el efecto es crear una suscripción en un punto determinado del árbol de temas. Si este punto del árbol de temas está representado por un objeto de tema administrativo, se comprueba un perfil de seguridad según el nombre de este objeto de tema.

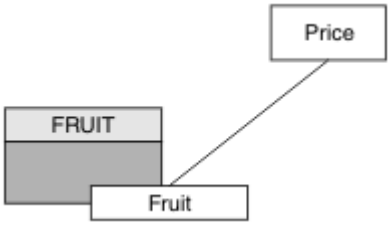


Figura 23. Ejemplo de acceso a objeto de tema

Tabla 82. Ejemplo de acceso a objeto de tema		
Tema	**Acceso de suscripción necesario**	**Objeto de tema**
Price	Ningún usuario	Ninguna
Price/Fruit	USER1	FRUIT

Defina un nuevo objeto de tema de la manera siguiente:

Procedimiento

1. Emita el mandato MQSC `DEF TOPIC(FRUIT) TOPICSTR('Price/Fruit')`.
2. Otorgue el acceso de la manera siguiente:

   - z/OS **z/OS** :

     Otorgue acceso a USER1 para suscribirse al tema `"Price/Fruit"` otorgando acceso de usuario al perfil `hlq.SUBSCRIBE.FRUIT`. Para ello, utilice los siguientes mandatos RACF:

     ```
     RDEFINE MXTOPIC hlq.SUBSCRIBE.FRUIT UACC(NONE)
     PERMIT hlq.SUBSCRIBE.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
     ```

   - Otras plataformas:

     Otorgue acceso a USER1 para suscribirse al tema `"Price/Fruit"` otorgando acceso de usuario al objeto FRUIT. Hágalo mediante el mandato de autorización para la plataforma:

     ULW **Sistemas Windows, UNIX and Linux**

     ```
     setmqaut -t topic -n FRUIT -p USER1 +sub
     ```

     IBM i **IBM i**

     ```
     GRTMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
     ```

Resultados

Cuando USER1 intenta suscribirse al tema `"Price/Fruit"`, el resultado es satisfactorio.

Cuando USER2 intenta suscribirse al tema `"Price/Fruit"` el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- z/OS En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

  ```
  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.FRUIT ...

  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
  ```

- ULW En otras plataformas, el siguiente suceso de autorización:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit"
  ```

- IBM i En IBMi, el siguiente suceso de autorización:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit"
  ```

Tenga en cuenta que es una ilustración de lo que verá, no de todos los campos.

Otorgar acceso a un usuario para suscribirse a un tema más profundamente en el árbol

Este tema es el segundo de una lista de tareas que indica cómo otorgar acceso a los temas por más de un usuario.

Antes de empezar

Este tema utiliza la configuración descrita en "Otorgar acceso a un usuario para suscribirse a un tema" en la página 480.

Acerca de esta tarea

Si el punto del árbol de temas en que la aplicación realiza la suscripción no está representada por un objeto de tema administrativo, suba en el árbol hasta localizar el objeto de tema administrativo padre más cercano. El perfil de seguridad se comprueba, basándose en el nombre de dicho objeto de tema.

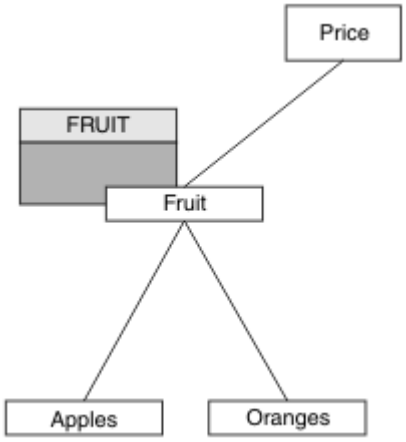


Figura 24. Ejemplo de otorgar acceso a un tema dentro de un árbol de temas

Tabla 83. Requisitos de acceso para los temas de ejemplo y los objetos de tema

Tema	Acceso de suscripción necesario	Objeto de tema
Price	Ningún usuario	Ninguna
Price/Fruit	USER1	FRUIT
Price/Fruit/Apples	USER1	
Price/Fruit/Oranges	USER1	

En la tarea anterior se otorgó acceso a USER1 para suscribirse al tema `"Price/Fruit"` otorgándole acceso al perfil `hlq.SUBSCRIBE.FRUIT` en z/OS y acceso de suscripción al perfil FRUIT en otras plataformas. Este perfil único también otorga acceso a USER1 para suscribirse a `"Price/Fruit/Apples"`, `"Price/Fruit/Oranges"` y `"Price/Fruit/#"`.

Cuando USER1 intenta suscribirse al tema `"Price/Fruit/Apples"`, el resultado es satisfactorio.

Cuando USER2 intenta suscribirse al tema `"Price/Fruit/Apples"` el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- En otras plataformas, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Apples"
```

Tenga en cuenta lo siguiente:

- Los mensajes que recibe en z/OS son idénticos a los recibidos en la tarea anterior ya que los mismos objetos de tema y perfiles controlan el acceso.
- El mensaje de suceso que recibe en otras plataformas es similar al recibido en la tarea anterior, pero la serie de tema real es diferente.

Otorgar acceso a otro usuario para suscribirse sólo al tema más profundamente en el árbol

Este tema es el tercero de una lista de tareas que indica cómo otorgar acceso para suscribirse a los temas por parte de más de un usuario.

Antes de empezar

Este tema utiliza la configuración descrita en “Otorgar acceso a un usuario para suscribirse a un tema más profundamente en el árbol” en la página 482.

Acerca de esta tarea

En la tarea anterior, se rechazó que USER2 tenga acceso al tema `"Price/Fruit/Apples"`. Este tema indica cómo otorgar acceso a dicho tema, pero no a otros temas.

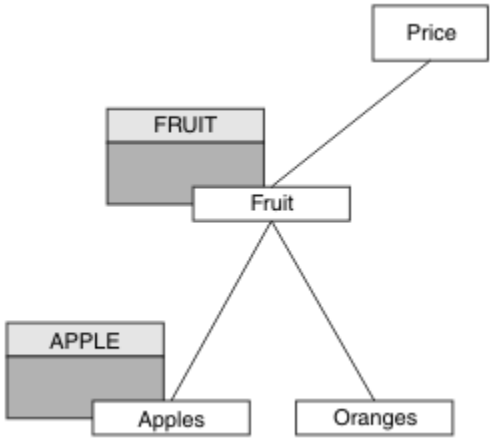


Figura 25. Otorgar acceso a temas específicos dentro de un árbol de temas

Tabla 84. Requisitos de acceso para los temas de ejemplo y los objetos de tema		
Tema	**Acceso de suscripción necesario**	**Objeto de tema**
Price	Ningún usuario	Ninguna
Price/Fruit	USER1	FRUIT
Price/Fruit/Apples	USER1 y USER2	APPLE
Price/Fruit/Oranges	USER1	

Defina un nuevo objeto de tema de la manera siguiente:

Procedimiento

1. Emita el mandato MQSC `DEF TOPIC(APPLE) TOPICSTR('Price/Fruit/Apples')`.
2. Otorgue el acceso de la manera siguiente:

- z/OS **z/OS** :

  En la tarea anterior, se otorgó acceso a USER1 para suscribirse al tema `"Price/Fruit/Apples"` otorgando al usuario acceso al perfil `hlq.SUBSCRIBE.FRUIT`.

  Este único perfil también otorgaba acceso a USER1 para suscribirse a `"Price/Fruit/Oranges"` `"Price/Fruit/#"` y este acceso permanece incluso al agregar el nuevo objeto de tema y los perfiles asociados al mismo.

  Otorgue acceso a USER2 para suscribirse al tema `"Price/Fruit/Apples"` otorgando acceso de usuario al perfil `hlq.SUBSCRIBE.APPLE`. Para ello, utilice los siguientes mandatos RACF:

  ```
  RDEFINE MXTOPIC hlq.SUBSCRIBE.APPLE UACC(NONE)
  PERMIT hlq.SUBSCRIBE.FRUIT APPLE(MXTOPIC) ID(USER2) ACCESS(ALTER)
  ```

- Otras plataformas:

  En la tarea anterior, se otorgó acceso a USER1 para suscribirse al tema `"Price/Fruit/Apples"` otorgando al usuario acceso de suscripción al perfil FRUIT.

  Este único perfil también otorgaba acceso a USER1 para suscribirse a `"Price/Fruit/Oranges"` y `"Price/Fruit/#"` y este acceso permanece incluso al agregar el nuevo objeto de tema y los perfiles asociados al mismo.

  Otorgue acceso a USER2 para suscribirse al tema `"Price/Fruit/Apples"` otorgando al usuario acceso de suscripción al perfil APPLE. Hágalo mediante el mandato de autorización para la plataforma:

  ULW **Sistemas Windows, UNIX and Linux**

  ```
  setmqaut -t topic -n APPLE -p USER2 +sub
  ```

  IBM i **IBM i**

  ```
  GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER2) AUT(*SUB)
  ```

Resultados

En z/OS, cuando USER1 intenta suscribirse al tema `"Price/Fruit/Apples"`, la primera comprobación de seguridad en el perfil `hlq.SUBSCRIBE.APPLE` falla, pero al subir en el árbol, el perfil `hlq.SUBSCRIBE.FRUIT` permite que USER1 se suscriba, de forma que la suscripción es satisfactoria y el código se envía a la llamada MQSUB. Sin embargo, se genera un mensaje RACF `ICH` para la primera comprobación:

```
ICH408I USER(USER1   ) ...
  hlq.SUBSCRIBE.APPLE ...
```

Cuando USER2 intenta suscribirse al tema `"Price/Fruit/Apples"` el resultado es satisfactorio porque la comprobación de seguridad pasa en el primer perfil.

Cuando USER2 intenta suscribirse al tema `"Price/Fruit/Oranges"` el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- z/OS En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

  ```
  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.FRUIT ...

  ICH408I USER(USER2   ) ...
    hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
  ```

- ULW En las plataformas Windows, UNIX and Linux, el suceso de autorización siguiente:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit/Oranges"
  ```

- IBM i En IBMi, el siguiente suceso de autorización:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
  UserIdentifier        USER2
  AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
  TopicString           "Price/Fruit/Oranges"
  ```

La desventaja de esta configuración es que, en z/OS, recibirá mensajes `ICH` adicionales en la consola. Puede evitarlo si protege el árbol de temas de forma diferente.

Cambio de control de acceso para evitar mensajes adicionales

Este tema es el cuarto de una lista de tareas que indica cómo otorgar acceso para suscribirse a temas por parte de más de un usuario y evitar mensajes RACF ICH408I adicionales en z/OS.

Antes de empezar

Este tema aumenta la configuración descrita en “Otorgar acceso a otro usuario para suscribirse sólo al tema más profundamente en el árbol” en la página 483 para que evite mensajes de error adicionales.

Acerca de esta tarea

En este tema se describe cómo puede otorgar acceso a los temas más profundos en el árbol y cómo eliminar el acceso al tema más abajo en el árbol cuando ningún usuario lo requiere.

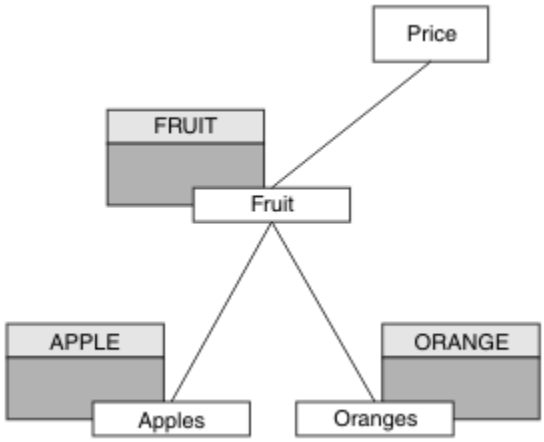


Figura 26. Ejemplo de otorgar control de acceso para evitar mensajes adicionales.

Defina un nuevo objeto de tema de la manera siguiente:

Procedimiento

1. Emita el mandato MQSC `DEF TOPIC(ORANGE) TOPICSTR('Price/Fruit/Oranges')`.
2. Otorgue el acceso de la manera siguiente:

- z/OS **z/OS** :

  Defina un nuevo perfil y agregue el acceso a dicho perfil y a los perfiles existentes. Para ello, utilice los siguientes mandatos RACF:

```
RDEFINE MXTOPIC hlq.SUBSCRIBE.ORANGE UACC(NONE)
PERMIT hlq.SUBSCRIBE.ORANGE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
PERMIT hlq.SUBSCRIBE.APPLE CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
```

- Otras plataformas:

  Configure el acceso equivalente con los mandatos de autorización de la plataforma:

  ULW **Sistemas Windows, UNIX and Linux**

```
setmqaut -t topic -n ORANGE -p USER1 +sub
setmqaut -t topic -n APPLE -p USER1 +sub
```

  IBM i **IBM i**

```
GRTMQAUT OBJ(ORANGE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
GRTMQAUT OBJ(APPLE) OBJTYPE(*TOPIC) USER(USER1) AUT(*SUB)
```

Resultados

En z/OS, cuando USER1 intenta suscribirse al tema `"Price/Fruit/Apples"`, la primera comprobación de seguridad del perfil `hlq.SUBSCRIBE.APPLE` es satisfactoria.

De forma similar, cuando USER2 intenta suscribirse al tema `"Price/Fruit/Apples"` el resultado es satisfactorio porque la comprobación de seguridad pasa en el primer perfil.

Cuando USER2 intenta suscribirse al tema `"Price/Fruit/Oranges"` el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- z/OS En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

```
ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.ORANGE ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.SUBSCRIBE.SYSTEM.BASE.TOPIC ...
```

- ULW En otras plataformas, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

- IBM i En IBMi, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_SUB_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      ORANGE, FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit/Oranges"
```

Otorgar acceso a un usuario para publicar en un tema

Este tema es el primero de una lista de tareas que indica cómo otorgar acceso para publicar temas por más de un usuario.

Acerca de esta tarea

En esta tarea se presupone que no existen objetos de temas administrativos en el lado derecho del árbol de temas, ni se han definido los perfiles para la publicación. La suposición utilizada es que los editores usan sólo la serie de tema.

Una aplicación puede publicar en un tema proporcionando un objeto de tema, o una serie de tema, o una combinación de ambos. Sea cual sea lo que seleccione la aplicación, el efecto es publicar en un punto determinado del árbol de temas. Si este punto del árbol de temas está representado por un objeto de tema administrativo, se comprueba un perfil de seguridad según el nombre de este objeto de tema. Por ejemplo:

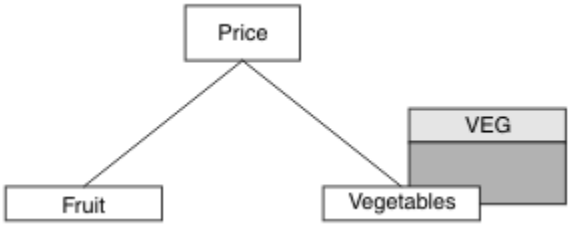


Figura 27. Otorgar acceso de publicación a un tema

Tabla 85. Ejemplo de requisitos de acceso de publicación

Tema	Acceso de publicación necesario	Objeto de tema
Price	Ningún usuario	Ninguna

Tabla 85. Ejemplo de requisitos de acceso de publicación (continuación)		
Tema	**Acceso de publicación necesario**	**Objeto de tema**
Price/Vegetables	USER1	VEG

Defina un nuevo objeto de tema de la manera siguiente:

Procedimiento

1. Emita el mandato MQSC `DEF TOPIC(VEG) TOPICSTR('Price/Vegetables')`.
2. Otorgue el acceso de la manera siguiente:
   - z/OS **z/OS** :

     Otorgue acceso a USER1 para publicar en el tema `"Price/Vegetables"` otorgando acceso de usuario al perfil `hlq.PUBLISH.VEG`. Para ello, utilice los siguientes mandatos RACF:

     ```
     RDEFINE MXTOPIC hlq.PUBLISH.VEG UACC(NONE)
     PERMIT hlq.PUBLISH.VEG CLASS(MXTOPIC) ID(USER1) ACCESS(UPDATE)
     ```

   - Otras plataformas:

     Otorgue acceso a USER1 para publicar en el tema `"Price/Vegetables"` otorgando acceso de usuario al perfil VEG. Hágalo mediante el mandato de autorización para la plataforma:

     ULW **Sistemas Windows, UNIX and Linux**

     ```
     setmqaut -t topic -n VEG -p USER1 +pub
     ```

     IBM i **IBM i**

     ```
     GRTMQAUT OBJ(VEG) OBJTYPE(*TOPIC) USER(USER1) AUT(*PUB)
     ```

Resultados

Cuando USER1 intenta publicar en el tema `"Price/Vegetables"`, el resultado es satisfactorio; es decir, la llamada MQOPEN es satisfactoria.

Cuando USER2 intenta publicar en el tema `"Price/Vegetables"`, el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- z/OS En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

  ```
  ICH408I USER(USER2   ) ...
    hlq.PUBLISH.VEG ...

  ICH408I USER(USER2   ) ...
    hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
  ```

- ULW En otras plataformas, el siguiente suceso de autorización:

  ```
  MQRC_NOT_AUTHORIZED
  ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
  UserIdentifier        USER2
  ```

```
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables"
```

- IBM i En IBMi, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables"
```

Tenga en cuenta que es una ilustración de lo que verá, no de todos los campos.

Otorgar acceso a un usuario para publicar en un tema más profundamente en el árbol

Este tema es el segundo de una lista de tareas que indica cómo otorgar acceso a los temas por más de un usuario.

Antes de empezar

Este tema utiliza la configuración descrita en “Otorgar acceso a un usuario para publicar en un tema” en la página 487.

Acerca de esta tarea

Si el punto del árbol de temas en que la aplicación realiza la publicación no está representada por un objeto de tema administrativo, suba en el árbol hasta localizar el objeto de tema administrativo padre más cercano. El perfil de seguridad se comprueba, basándose en el nombre de dicho objeto de tema.

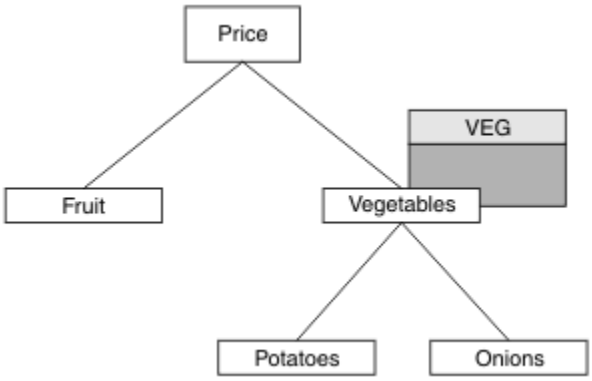


Figura 28. Otorgar acceso de publicación a un tema dentro de un árbol de temas

Tabla 86. Ejemplo de requisitos de acceso de publicación

Tema	Acceso de suscripción necesario	Objeto de tema
Price	Ningún usuario	Ninguna
Price/Vegetables	USER1	VEG
Price/ Vegetables/ Potatoes	USER1	
Price/ Vegetables/ Onions	USER1	

En la tarea anterior se otorgó acceso a USER1 al tema de publicación "Price/Vegetables/Potatoes" al otorgarle acceso al perfil hlq.PUBLISH.VEG en z/OS o acceso de publicación al perfil VEG en otras plataformas. Este perfil único también otorga acceso a USER1 para publicar en "Price/Vegetables/Onions".

Cuando USER1 intenta publicar en el tema "Price/Vegetables/Potatoes", el resultado es satisfactorio; es decir, la llamada MQOPEN es satisfactoria.

Cuando USER2 intenta suscribirse al tema "Price/Vegetables/Potatoes", el resultado es anómalo; es decir, la llamada MQOPEN falla con un mensaje MQRC_NOT_AUTHORIZED, junto con:

- En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.VEG ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- En otras plataformas, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      VEG, SYSTEM.BASE.TOPIC
TopicString           "Price/Vegetables/Potatoes"
```

Tenga en cuenta lo siguiente:

- Los mensajes que recibe en z/OS son idénticos a los recibidos en la tarea anterior ya que los mismos objetos de tema y perfiles controlan el acceso.
- El mensaje de suceso que recibe en otras plataformas es similar al recibido en la tarea anterior, pero la serie de tema real es diferente.

Otorgar acceso para publicar y suscribir

Este tema es el último de una lista de tareas que indica cómo otorgar acceso para publicar y suscribirse a temas por más de un usuario.

Antes de empezar

Este tema utiliza la configuración descrita en “Otorgar acceso a un usuario para publicar en un tema más profundamente en el árbol” en la página 489.

Acerca de esta tarea

En una tarea anterior se otorgó acceso a USER1 para suscribirse al tema "Price/Fruit". Este tema explica cómo otorgar acceso a dicho usuario para publicar en dicho tema.

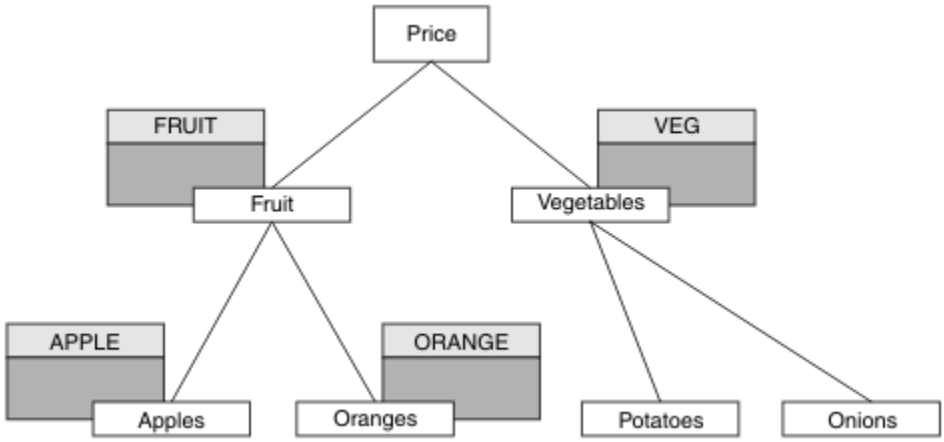


Figura 29. Otorgar acceso para publicar y suscribir

Tabla 87. Ejemplo de requisitos de acceso de publicación y suscripción

Tema	Acceso de suscripción necesario	Acceso de publicación necesario	Objeto de tema
Price	Ningún usuario	Ningún usuario	Ninguna
Price/Fruit	USER1	USER1	FRUIT
Price/Fruit/Apples	USER1 y USER2		APPLE
Price/Fruit/Oranges	USER1		ORANGE

Procedimiento

Otorgue el acceso de la manera siguiente:

- z/OS **z/OS** :

  En una tarea anterior se otorgó acceso a USER1 para suscribirse al tema `"Price/Fruit"` otorgando al usuario acceso al perfil `hlq.SUBSCRIBE.FRUIT`.

  Para publicar en el tema `"Price/Fruit"`, otorgue acceso a USER1 al perfil `hlq.PUBLISH.FRUIT`. Para ello, utilice los siguientes mandatos RACF:

  ```
  RDEFINE MXTOPIC hlq.PUBLISH.FRUIT UACC(NONE)
  PERMIT hlq.PUBLISH.FRUIT CLASS(MXTOPIC) ID(USER1) ACCESS(ALTER)
  ```

- Otras plataformas:

  Otorgue acceso a USER1 para publicar en el tema `"Price/Fruit"` otorgando acceso de publicación de usuario al objeto FRUIT. Hágalo mediante el mandato de autorización para la plataforma:

  ULW **Sistemas Windows, UNIX and Linux**

  ```
  setmqaut -t topic -n FRUIT -p USER1 +pub
  ```

IBM i

```
GRTMQAUT OBJ(FRUIT) OBJTYPE(*TOPIC) USER(USER1) AUT(*PUB)
```

Resultados

En z/OS, cuando USER1 intenta publicar en el tema `"Price/Fruit"` se pasa la comprobación de seguridad en la llamada MQOPEN.

Cuando USER2 intenta publicar en el tema `"Price/Fruit"` el resultado es anómalo con un mensaje `MQRC_NOT_AUTHORIZED`, junto con:

- En z/OS, se ven los mensajes siguientes en la consola que muestran la vía de acceso de seguridad completa a través del árbol de temas que se ha intentado:

```
ICH408I USER(USER2   ) ...
  hlq.PUBLISH.FRUIT ...

ICH408I USER(USER2   ) ...
  hlq.PUBLISH.SYSTEM.BASE.TOPIC ...
```

- En las plataformas Windows, UNIXy Linux, el suceso de autorización siguiente:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

- En IBMi, el siguiente suceso de autorización:

```
MQRC_NOT_AUTHORIZED
ReasonQualifier      MQRQ_OPEN_NOT_AUTHORIZED
UserIdentifier        USER2
AdminTopicNames      FRUIT, SYSTEM.BASE.TOPIC
TopicString           "Price/Fruit"
```

Tras el conjunto completo de estas tareas, ofrece a USER1 y USER2 las autorizaciones de acceso siguientes para publicar y suscribirse a los temas que se listan:

Tabla 88. Lista completa de las autorizaciones de acceso resultantes de ejemplos de seguridad

Tema	Acceso de suscripción necesario	Acceso de publicación necesario	Objeto de tema
Price	Ningún usuario	Ningún usuario	Ninguna
Price/Fruit	USER1	USER1	FRUIT
Price/Fruit/Apples	USER1 y USER2		APPLE
Price/Fruit/Oranges	USER1		ORANGE
Price/Vegetables		USER1	VEG

Tabla 88. Lista completa de las autorizaciones de acceso resultantes de ejemplos de seguridad (continuación)

Tema	Acceso de suscripción necesario	Acceso de publicación necesario	Objeto de tema
Price/ Vegetables/ Potatoes			
Price/ Vegetables/ Onions			

Donde tenga requisitos distintos de acceso de seguridad a distintos niveles dentro del árbol de temas, una planificación cuidadosa asegura que no recibirá avisos de seguridad improcedentes en el registro de la consola de z/OS. La configuración de seguridad en el nivel correcto dentro del árbol evita mensajes de seguridad engañosos.

Seguridad de suscripción

MQSO_ALTERNATE_USER_AUTHORITY

El campo AlternateUserId contiene un identificador de usuario para utilizarlo para validar esta llamada MQSUB. La llamada solo será satisfactoria si este AlternateUserId tiene autorización para suscribirse al tema con las opciones de acceso especificadas, independientemente de si el identificador del usuario bajo el que está ejecutándose la aplicación tiene autorización para ello.

MQSO_SET_IDENTITY_CONTEXT

La suscripción debe utilizar la señal de contabilidad y los datos de identidad de la aplicación suministrados en los campos PubAccountingToken y PubApplIdentityData.

Si se especifica esta opción, se realiza la misma comprobación de autorización que si se accediera a la cola de destino mediante una llamada MQOPEN con MQOO_SET_IDENTITY_CONTEXT, excepto en el caso en que se utilice también la opción MQSO_MANAGED, en cuyo caso no hay comprobación de autorización en la cola de destino.

Si no se especifica esta opción, las publicaciones enviadas a este suscriptor tienen información de contexto predeterminada asociada a ellos, de la manera siguiente:

Tabla 89. Información de contexto de publicación predeterminado

Campo de MQMD	Valor utilizado
UserIdentifier	El ID de usuario asociado a la suscripción (vea el campo SUBUSER en DISPLAY SBSTATUS) cuando se realizó la publicación.
AccountingToken	Determinado por el entorno si es posible; en caso contrario, establecido en MQACT.
ApplIdentityData	Establecido en blancos.

Esta opción sólo es válida con MQSO_CREATE y MQSO_ALTER. Si se utiliza con MQSO_RESUME, los campos PubAccountingToken y PubApplIdentityData se ignoran, por lo que esta opción no tiene ningún efecto.

Si se modifica una suscripción sin utilizar esta opción en la que previamente la suscripción había facilitado información de contexto de identidad, se genera información del contexto predeterminado para la suscripción modificada.

Si una suscripción que permite que distintos ID de usuario la utilicen con la opción MQSO_ANY_USERID, se reanuda con un ID de usuario diferente, se genera contexto de identidad predeterminado para el nuevo ID de usuario que es propietario ahora de la suscripción y las publicaciones posteriores se entregan conteniendo el nuevo contexto de identidad.

AlternateSecurityId

Este es un identificador de seguridad que se transfiere con el AlternateUserId al servicio de autorizaciones para permitir que se realicen las comprobaciones de autorización correspondientes. AlternateSecurityId sólo se utiliza si se especifica MQSO_ALTERNATE_USER_AUTHORITY y el campo AlternateUserId no está completamente en blanco hasta el primer carácter nulo o el final del campo.

Opción de suscripción MQSO_ANY_USERID

Cuando se especifica MQSO_ANY_USERID, la identidad del suscriptor no está restringida a un ID de usuario único. Esto permite que cualquier usuario modifique o reanude la suscripción cuando disponga de la autoridad adecuada. Sólo puede tener la suscripción un único usuario a la vez. Un intento de reanudar el uso de una suscripción utilizada actualmente por otra aplicación hará que falle la llamada con MQRC_SUBSCRIPTION_IN_USE.

Para añadir esta opción a una suscripción existente, la llamada MQSUB (utilizando MQSO_ALTER) debe proceder del mismo ID de usuario que la suscripción original.

Si una llamada MQSUB hace referencia a una suscripción existente con MSQO_ANY_USERID establecido y el ID de usuario difiere de la suscripción original, la llamada sólo será satisfactoria si el nuevo ID de usuario tiene autorización para suscribirse al tema. Tras la finalización satisfactoria, las futuras publicaciones de este suscriptor se colocarán en la cola del suscriptor con el nuevo ID de usuario establecido en la publicación.

MQSO_FIXED_USERID

Cuando se especifica MQSO_FIXED_USERID, sólo un ID de usuario propietario puede modificar o reanudar la suscripción. Este ID de usuario es el último ID de usuario para modificar la suscripción que estableció esta opción, eliminando así la opción MQSO_ANY_USERID, o si se ha llevado a cabo ninguna modificación, es el ID de usuario que ha creado la suscripción.

Si un verbo MQSUB hace referencia a una suscripción existente con MQSO_ANY_USERID establecido y modifica la suscripción (utilizando MQSO_ALTER) para utilizar la opción MQSO_FIXED_USERID, el ID de usuario de la suscripción se ha fijado ahora en este ID de usuario nuevo. La llamada sólo es satisfactoria si el nuevo ID de usuario tiene autoridad para suscribirse al tema.

Si un ID de usuario distinto del registrado como propietario de una suscripción intenta reanudar o modificar una suscripción MQSO_FIXED_USERID, la llamada fallará con MQRC_IDENTITY_MISMATCH. El ID de usuario propietario de una suscripción se puede ver mediante el mandato DISPLAY SBSTATUS.

Si no se especifica MQSO_ANY_USERID ni MQSO_FIXED_USERID, el valor predeterminado es MQSO_FIXED_USERID.

Seguridad de publicación/suscripción entre gestores de colas

Los mensajes internos de publicación/suscripción como, por ejemplo, las suscripciones y publicaciones proxy, se colocan en colas de sistema de publicación/suscripción mediante las reglas normales de seguridad de canal. La información y diagramas de este tema resaltan los diversos procesos y los ID de usuario implicados en la entrega de estos mensajes.

Control de acceso local

El acceso a temas para publicación y suscripciones se rige por las definiciones de seguridad locales y las reglas que se describen en Seguridad de publicación/suscripción. En z/OS, no hay ningún objeto de tema local necesario para establecer control de acceso. Tampoco hay ningún tema local necesario para el control de accesos en otras plataformas. Los administradores pueden optar por aplicar el control de acceso a objetos de tema de clúster, independientemente de si existen en el clúster todavía.

Los administradores de sistema son responsables del control de acceso en su sistema local. Deben confiar en los administradores de otros miembros de la jerarquía o colectivos de clúster de ser responsables de su política de control de acceso. Como el control de acceso está definido para cada máquina por separado, es probable que sea un contratiempo si se necesita un control de nivel muy preciso. Puede que no sea necesario imponer ningún control de acceso, o que pueda definirse el control de acceso en los objetos de alto nivel del árbol de temas. Puede definirse un control de acceso de nivel más preciso para cada subdivisión del espacio de nombres de temas.

Realizar una suscripción proxy

La confianza de una organización al conectar su gestor de colas al gestor de colas del usuario se confirma por medios normales de autenticación de canal. Si a la organización de confianza también se le permite realizar una publicación/suscripción distribuida, se efectúa una comprobación de autoridad. La comprobación se realiza cuando el canal coloca un mensaje en una cola de publicación/suscripción distribuida. Por ejemplo, si se coloca un mensaje en la cola `SYSTEM.INTER.QMGR.CONTROL` . El ID de usuario para la comprobación de autorización de cola depende de los valores de PUTAUT del canal receptor. Por ejemplo, el ID de usuario del canal, `MCAUSER`, el contexto del mensaje, según el valor y plataforma. Para obtener más información sobre la seguridad de canal, consulte Seguridad de canal.

Las suscripciones proxy se realizan con el ID de usuario del agente de publicación/suscripción distribuido en el gestor de colas remoto. Por ejemplo, `QM2` en Figura 30 en la página 495. Entonces se otorga acceso al usuario fácilmente a los perfiles de objeto de tema locales, porque dicho ID de usuario está definido en el sistema y, por consiguiente, no hay conflictos de dominio.

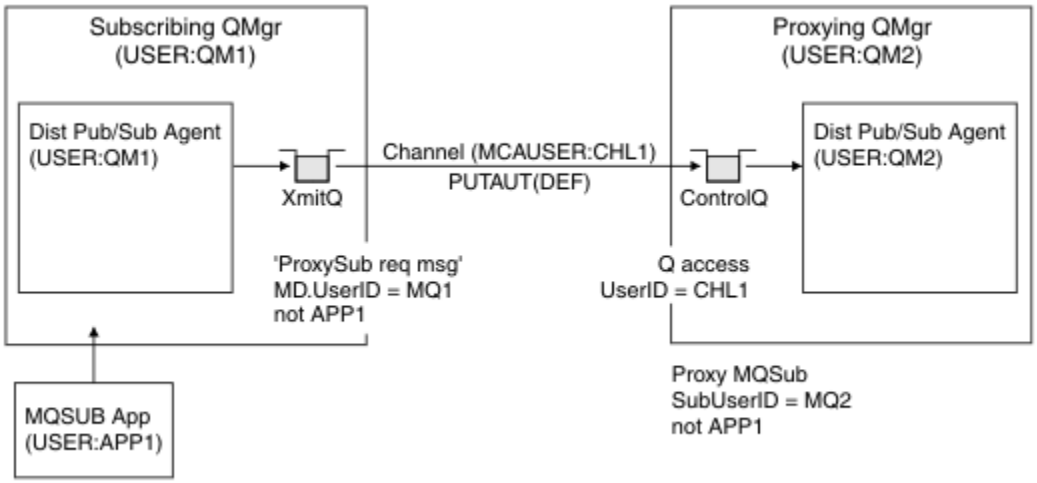


Figura 30. Seguridad de suscripción proxy, realizar una suscripción

Devolver publicaciones remotas

Cuando se crea una publicación en el gestor de colas de publicación, una copia de la publicación se crea para cualquier suscripción proxy. El contexto de la publicación copiada contiene el contexto del ID de usuario que hizo la suscripción; `QM2` en Figura 31 en la página 496. La suscripción proxy se crea con una cola de destino que es una cola remota, por lo que el mensaje de publicación se resuelve en una cola de transmisión.

De confianza para una organización al conectar su gestor de colas, QM2, a otro gestor de colas, QM1, se confirma por medios normales de autenticación de canal. Si se permite a la organización de confianza realizar la publicación/suscripción distribuida, se lleva a cabo una comprobación de autorización cuando el canal coloca el mensaje de publicación en la cola de publicación de la publicación/suscripción distribuida SYSTEM.INTER.QMGR.PUBS. El ID de usuario de la comprobación de autorización de cola depende del valor de PUTAUT del canal receptor (por ejemplo, el ID de usuario del canal, MCAUSER, contexto de mensaje y otros, en función del valor y la plataforma). Para obtener más información sobre la seguridad de canal, consulte Seguridad de canal.

Cuando el mensaje de publicación llega al gestor de colas de suscripción, se realiza otro MQPUT en el tema bajo la autorización de dicho gestor de colas y el contexto que incluye el mensaje es sustituido por el contexto de cada uno de los suscriptores locales a medida que se les entrega el mensaje.

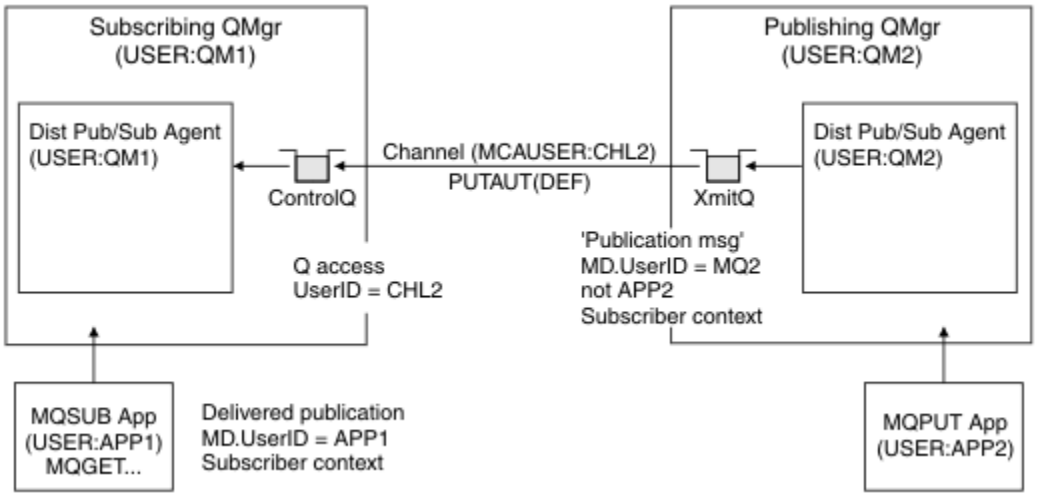


Figura 31. Seguridad de suscripción proxy, reenviando publicaciones

En un sistema en el que poco se ha tenido en cuenta en relación con la seguridad, los procesos de publicación/suscripción distribuidos probablemente se estén ejecutando bajo un ID de usuario del grupo mqm, el parámetro MCAUSER en un canal está en blanco (el valor predeterminado), y los mensajes se entregan a las diversas colas del sistema, según sea necesario. El sistema no seguro facilita la configuración de una prueba de concepto para demostrar la publicación/suscripción distribuida.

En un sistema donde la seguridad es considerada más seriamente, estos mensajes internos están sujetos a los controles de seguridad igual que cualquier mensaje que pase a través del canal.

Si el canal está configurado con un carácter no blanco MCAUSER y un valor PUTAUT que especifica que MCAUSER debe comprobarse, entonces debe concederse al MCAUSER en cuestión acceso a las colas SYSTEM.INTER.QMGR.*. Si hay varios gestores de colas remotos diferentes, con canales que se ejecutan con distintos ID de MCAUSER, es necesario otorgar a todos los ID de usuario acceso a las colas SYSTEM.INTER.QMGR.*. Pueden aparecer canales que se ejecutan con ID de MCAUSER diferentes cuando, por ejemplo, varias conexiones jerárquicas se configuran en un único gestor de colas.

Si el canal está configurado con un valor PUTAUT que especifica que se utiliza el contexto del mensaje, entonces el acceso a la colas SYSTEM.INTER.QMGR.* se comprueba basándose en el ID de usuario dentro del mensaje interno. Dado que todos estos mensajes se transfieren con el ID de usuario del agente de publicación/suscripción distribuido del gestor de colas que envía el mensaje interno o el mensaje de publicación (consulte Figura 31 en la página 496), un conjunto de ID de usuario para otorgar acceso a las diversas colas de sistema no es demasiado grande (uno por cada gestor de colas remoto), si desea configurar la seguridad de publicación/suscripción distribuida de esta manera. Aún tiene las mismas cuestiones que siempre tiene la seguridad de contexto de canal; las de los diferentes dominios de ID de usuario y el hecho de que el ID de usuario en el mensaje podría no estar definido en el sistema receptor. Sin embargo, es una manera perfectamente aceptable de ejecutarlo si es necesario.

z/OS La sección Seguridad de colas del sistema proporciona una lista de colas y el acceso que se necesita para configurar de forma segura el entorno de publicación/suscripción distribuida. Si los

mensajes internos o publicaciones no se transfieren debido a violaciones de seguridad, el canal escribe un mensaje en el registro de la forma normal y los mensajes se pueden enviar a la cola de mensajes no entregados de acuerdo con el proceso de errores de canal normal.

Todos los mensajes entre gestores de colas a efectos de publicación/suscripción distribuida se ejecutan utilizando la seguridad de canal normal.

Para obtener información sobre la restricción de las publicaciones y suscripciones proxy en el nivel de tema, consulte Seguridad de publicación/suscripción.

Utilización de los ID de usuario predeterminados con una jerarquía de gestores de colas

Si tiene una jerarquía de gestores de colas que se ejecutan en plataformas diferentes y utilizan los ID de usuario predeterminados, tenga en cuenta que estos ID de usuario predeterminados difieren entre plataformas y es posible que no sean conocidos en la plataforma de destino. Como resultado, un gestor de colas que se ejecuta en una plataforma rechaza los mensajes recibidos de los gestores de colas de otras plataformas con el código de razón `MQRC_NOT_AUTHORIZED`.

Para evitar que se rechacen mensajes, como mínimo, las autorizaciones siguientes deben añadirse a los ID de usuario predeterminados utilizados en otras plataformas:

- Autorización *PUT *GET en las colas SYSTEM.BROKER. colas
- Autorización *PUB *SUB en los temas SYSTEM.BROKER. temas
- Autorización *ADMCRT *ADMDLT *ADMCHG en la cola SYSTEM.BROKER.CONTROL.QUEUE.

Los ID de usuario predeterminados con una jerarquía de gestores de colas son los siguientes:

Plataforma	**ID de usuario predeterminado**
Windows	`MUSR_MQADMIN`
Sistemas UNIX and Linux	`mqm`
IBM i	`QMQM`
z/OS	El ID de usuario del espacio de direcciones del iniciador de canal

Cree y otorgue acceso al ID de usuario 'qmqm' si está conectado jerárquicamente a un gestor de colas en IBM i for Queue Managers en las plataformas Windows, UNIX, Linux y z/OS.

Cree y otorgue acceso al ID de usuario 'mqm' si está conectado jerárquicamente con un gestor de colas en Windows, UNIX o Linux for Queue Managers en las plataformas IBM i y z/OS.

Cre y otorgue acceso de usuario al ID de usuario del espacio de direcciones de iniciador de canal de z/OS si está conectado jerárquicamente a un gestor de colas en z/OS for Queue Managers en las plataformas Windows, UNIX, Linux y IBM i.

Los ID de usuario pueden distinguir entre mayúsculas y minúsculas. El gestor de colas de origen (si se trata de los sistemas IBM i, Windows, UNIX o Linux) fuerza que el ID de usuario se escriba totalmente en mayúsculas. El gestor de colas receptor (si se trata de los sistemas Windows, UNIX o Linux) fuerza que el ID de usuario se escriba totalmente en minúsculas. Por lo tanto, todos los ID de usuario creados en los sistemas UNIX and Linux deben crearse en minúsculas. Si se ha instalado una salida de mensaje, no se fuerza al ID de usuario a escribirse en mayúsculas o minúsculas. Hay que tener cuidado para comprender cómo la salida de mensajes procesa el ID de usuario.

Para evitar posibles problemas con la conversión de los ID de usuario:

- En los sistemas UNIX, Linux, and Windows, asegúrese de que los ID de usuario se han especificado en minúsculas.
- En IBM i y z/OS, asegúrese de que los ID de usuario se especifican en mayúsculas.

V 9.1.0 Seguridad de IBM MQ Console y REST API

La seguridad de IBM MQ Console y REST API se configura añadiendo la configuración del servidor mqweb en el archivo `mqwebuser.xml`.

Acerca de esta tarea

Puede realizar un seguimiento de las acciones de usuario y auditar el uso de IBM MQ Console y REST API examinando los archivos de registro del servidor mqweb.

Los usuarios de IBM MQ Console y REST API se pueden autenticar utilizando:

- Un registro básico
- Un registro LDAP
- Un registro de sistema operativo
- SAF en z/OS
- Cualquier otro tipo de registro soportado por WebSphere Liberty

Los roles se pueden asignar a usuarios IBM MQ Console y a usuarios REST API para determinar el nivel de acceso que se les otorga a los objetos de IBM MQ. Por ejemplo, para realizar la mensajería, los usuarios deben tener asignado el rol `MQWebUser`. Si desea más información sobre los roles disponibles, consulte "Roles en IBM MQ Console y REST API" en la página 509.

Una vez asignado un rol a un usuario, pueden utilizarse varios métodos para autenticar el usuario. Con IBM MQ Console, los usuarios pueden iniciar una sesión con un nombre de usuario y una contraseña, o pueden utilizar la autenticación de certificados de cliente. Con la REST API, los usuarios pueden utilizar la autenticación HTTP básica, la autenticación basada en señal o la autenticación de certificado de cliente.

Procedimiento

1. Defina el registro de usuarios para autenticar los usuarios y asigne a cada usuario o grupo un rol para que puedan utilizar IBM MQ Console o REST API. Para obtener más información, consulte "Configuración de usuarios y roles" en la página 499
2. Elija cómo se autentican los usuarios de IBM MQ Console con el servidor mqweb. No es necesario que utilice el mismo método para todos los usuarios:
   - Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional la hora de caducidad de la señal LTPA. Si desea más información, consulte Configuración del intervalo de caducidad de señal LTPA.
   - Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.
3. Elija cómo se autentican los usuarios de REST API con el servidor mqweb. No es necesario que utilice el mismo método para todos los usuarios:
   - Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte "Utilización de la autenticación básica HTTP con REST API" en la página 514.
   - Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado

durante una cantidad de tiempo definida. Para obtener más información, consulte "Utilización de la autenticación basada en señal con la API REST" en la página 516.

Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Sin embargo, si ha habilitado las conexiones HTTP, puede permitir que se emita una señal LTPA para que se utilice una conexión HTTPS para una conexión HTTP. Para obtener más información, consulte Configuración de la señal LTPA.

- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

4. Opcional: Configure Cross Origin Resource Sharing (CORS) para REST API.

   De forma predeterminada, un navegador web no permite que scripts como, por ejemplo, JavaScript, invoquen REST API cuando no provienen del mismo origen que REST API. Es decir, las solicitudes entre orígenes no están habilitadas. Puede configurar Cross Origin Resource Sharing (CORS) para permitir las solicitudes entre orígenes a partir de los URL especificados. Para obtener más información, consulte "Configuración de CORS para REST API" en la página 518.

5. Opcional: Configure la validación de la cabecera de host para la IBM MQ Console y la REST API.

   Puede configurar la validación de cabecera de host y crear una lista de elementos permitidos de nombres de host y puertos para asegurarse de que IBM MQ Console y REST APIsólo procesan las solicitudes que contienen cabeceras de host específicas. Para obtener más información, consulte "Configurando la validación de la cabecera de host para la IBM MQ Console y la REST API" en la página 519.

V 9.1.0 Configuración de usuarios y roles

Para poder utilizar IBM MQ Console o REST API, los usuarios deben autenticarse en un registro de usuarios, definido en el servidor mqweb.

Acerca de esta tarea

Los usuarios autenticados deben ser miembros de uno de los grupos que autoriza el acceso a las prestaciones de IBM MQ Console y REST API. De forma predeterminada, el registro de usuarios no contiene ningún usuario; estos deben añadirse editando el archivo `mqwebuser.xml`.

Cuando se configuran usuarios y grupos, primero debe configurar un registro de usuarios con el que autenticar los usuarios y los grupos. Este registro de usuarios está compartido entre la IBM MQ Console y la REST API. Puede controlar si los usuarios y grupos tienen acceso a IBM MQ Console, REST API, o a ambos, mediante la configuración de roles para los usuarios y grupos.

Después de configurar el registro de usuarios, puede configurar roles para los usuarios y grupos para otorgarles autorizaciones. Existen varios roles disponibles, incluyendo roles específicos al uso de REST API para Managed File Transfer. Cada rol otorga un nivel de acceso diferente. Para obtener más información, consulte "Roles en IBM MQ Console y REST API" en la página 509.

Se proporcionan varios archivos XML de ejemplo con el servidor mqweb para simplificar la configuración de usuarios y grupos. Es posible que los usuarios que están familiarizados con la configuración de la seguridad en WebSphere Liberty (WLP) prefieran no utilizar los ejemplos. WLP proporciona otras funciones de autorización además de las documentadas aquí.

Procedimiento

- Configure usuarios y grupos con un registro básico utilizando el archivo `basic_registry.xml`.

  Los nombres de usuario y las contraseñas en el registro se utilizan para autenticar y autorizar usuarios de IBM MQ Console y REST API.

  Para configurar un registro básico utilizando el archivo de ejemplo `basic_registry.xml`, consulte "Configuración de un registro básico para IBM MQ Console y REST API" en la página 501.

- Configure usuarios y grupos con un registro LDAP utilizando el archivo `ldap_registry.xml`.

  Los nombres de usuario y las contraseñas en el registro LDAP se utilizan para autenticar y autorizar usuarios de IBM MQ Console y REST API.

  Para configurar un registro LDAP utilizando el archivo de ejemplo `ldap_registry.xml`, consulte "Configuración de un registro LDAP para la IBM MQ Console y la REST API" en la página 505.

- ULW

  Configure usuarios y grupos con un registro de sistema operativo local utilizando el archivo `local_os_registry.xml`.

  Los nombres de usuario y las contraseñas en el registro del sistema operativo se utilizan para autenticar y autorizar usuarios de IBM MQ Console y REST API.

  Para configurar un registro de SO local utilizando el archivo de ejemplo `local_os_registry.xml`, consulte "Configuración de un registro de SO local para la IBM MQ Console y la REST API" en la página 503.

- z/OS

  Configure usuarios y grupos con la interfaz SAF (System Authorization Facility) en z/OS utilizando el archivo `zos_saf_registry.xml`.

  Los perfiles RACF, u otro producto de seguridad, se utilizan para otorgar a los usuarios y grupos acceso a los roles. Los nombres de usuario y las contraseñas en la base de datos RACF se utilizan para autenticar y autorizar usuarios de IBM MQ Console y REST API.

  Para configurar la interfaz SAF utilizando el archivo de ejemplo `zos_saf_registry.xml`, consulte "Configuración de un registro SAF para la IBM MQ Console y REST API" en la página 506.

- Inhabilite la seguridad, incluyendo la capacidad de acceder a la IBM MQ Console, o a la REST API, utilizando el archivo `no_security.xml`.

Qué hacer a continuación

Elija cómo se autentican los usuarios:

Opciones de autenticación de IBM MQ Console

- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración del intervalo de caducidad de señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

Opciones de autenticación de REST API

- Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte "Utilización de la autenticación básica HTTP con REST API" en la página 514.
- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. Para obtener más información, consulte "Utilización de la autenticación basada en señal con la API REST" en la página 516. Puede configurar el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración de la señal LTPA.

- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

V 9.1.0 Configuración de un registro básico para IBM MQ Console y REST API

Puede configurar un registro básico dentro del archivo `mqwebuser.xml`. Los nombres de usuario, contraseñas y roles del archivo xml se utiliza para autenticar y autorizar usuarios de IBM MQ Console y REST API.

Antes de empezar

- Al configurar usuarios dentro del registro básico, debe asignar un rol a cada usuario. Cada rol proporciona distintos niveles de privilegio para acceder a IBM MQ Console y REST API, y determina el contexto de seguridad que se utiliza cuando se intenta una operación permitida. Tendrá que comprender estos roles antes de poder configurar el registro básico. Si desea más información sobre cada uno de los roles, consulte "Roles en IBM MQ Console y REST API" en la página 509.
- Para completar esta tarea, debe ser un usuario con privilegios suficientes para editar el archivo `mqwebuser.xml`:
  - z/OS En z/OS, debe tener acceso de escritura en el archivo `mqwebuser.xml`.
  - Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Procedimiento

1. Copie el archivo XML de ejemplo `basic_registry.xml` de una de las vías de acceso siguientes:
   - ULW En UNIX, Linux, and Windows: *MQ_INSTALLATION_PATH* `/web/mq/samp/configuration`
   - z/OS En z/OS: `PathPrefix /web/mq/samp/configuration`

     donde `PathPrefix` es la vía de acceso de instalación de IBM MQ Unix System Services Components.
2. Coloque el archivo de ejemplo en el directorio adecuado:
   - ULW

     en UNIX, Linux, and Windows: *MQ_DATA_PATH*`/web/installations/`*installationName*`/servers/mqweb`
   - z/OS

     en z/OS: *WLP_user_directory*`/servers/mqweb`

     donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
3. Opcional: Si ha cambiado valores de configuración en `mqwebuser.xml`, cópielos en el archivo de ejemplo.
4. Suprima el archivo `mqwebuser.xml` existente y cambie el nombre del archivo de ejemplo a `mqwebuser.xml`.
5. Edite el nuevo archivo `mqwebuser.xml` para añadir usuarios y grupos dentro de los códigos **basicRegistry**.

   Tenga en cuenta que cualquier usuario con el rol `MQWebUser` solo puede realizar las operaciones otorgadas al ID de usuario en el gestor de colas. Por tanto, el ID de usuario definido en el registro ha

de tener un ID de usuario idéntico en el sistema en el que está instalado IBM MQ. Estos ID de usuario tienen que coincidir en mayúsculas y minúsculas o la correlación entre ellos puede fallar.

Para obtener más información sobre cómo configurar registros de usuarios básicos, consulte Configuración de un registro de usuarios básico en Liberty en la documentación de WebSphere Liberty.

6. Asigne roles a usuarios y grupos editando el archivo `mqwebuser.xml`:

   Existen varios roles disponibles que autorizan a usuarios y grupos a utilizar IBM MQ Console, y REST API. Cada rol otorga un nivel de acceso diferente. Para obtener más información, consulte "Roles en IBM MQ Console y REST API" en la página 509.

   - Para asignar roles y otorgar acceso a IBM MQ Console, añada los usuarios y los grupos entre las etiquetas **security-role** adecuadas en las etiquetas **<enterpriseApplication id="com.ibm.mq.console">**.
   - Para asignar roles y otorgar acceso a REST API, añada los usuarios y los grupos entre las etiquetas **security-role** adecuadas en las etiquetas **<enterpriseApplication id="com.ibm.mq.rest">**.

   Para obtener ayuda con el formato de la información de usuarios y grupos dentro de las etiquetas **security-role**, consulte los ejemplos.

7. Si ha proporcionado las contraseñas de los usuarios en `mqwebuser.xml`, debería codificar dichas contraseñas para hacerlas más seguras con el mandato **securityUtility encoding** proporcionado por WebSphere Liberty. Para obtener más información, consulte Liberty: MandatosecurityUtility en la documentación del producto WebSphere Liberty .

Ejemplo

En el siguiente ejemplo, se otorga acceso al grupo `MQWebAdminGroup` a IBM MQ Console con el rol `MQWebAdmin`. Al usuario `reader` se le otorga acceso con el rol `MQWebAdminRO` y al usuario `guest` se le otorga acceso con el rol `MQWebUser`:

```
<enterpriseApplication id="com.ibm.mq.console">
        <application-bnd>
            <security-role name="MQWebAdmin">
                <group name="MQWebAdminGroup" realm="defaultRealm"/>
            </security-role>
            <security-role name="MQWebAdminRO">
                <user name="reader" realm="defaultRealm"/>
            </security-role>
            <security-role name="MQWebUser">
                <user name="guest" realm="defaultRealm"/>
            </security-role>
        </application-bnd>
    </enterpriseApplication>
```

En el siguiente ejemplo, a los usuarios `reader` y `guest` se les otorga acceso a IBM MQ Console. Al usuario `user` se le otorga acceso a REST API, y a cualquier usuario del grupo `MQAdmin` se le otorga acceso a IBM MQ Console y REST API. Al usuario `mftadmin` se le otorga acceso a REST API para MFT :

```
<enterpriseApplication id="com.ibm.mq.console">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="MQAdmin" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebAdminRO">
            <user name="reader" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebUser">
            <user name="guest" realm="defaultRealm"/>
        </security-role>
    </application-bnd>
</enterpriseApplication>

<enterpriseApplication id="com.ibm.mq.rest">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="MQAdmin" realm="defaultRealm"/>
        </security-role>
        <security-role name="MQWebUser">
```

```
            <user name="user" realm="defaultRealm"/>
        </security-role>
        <security-role name="MFTWebAdmin">
            <user name="mftadmin" realm="defaultRealm"/>
        </security-role>
    </application-bnd>
</enterpriseApplication>
```

Qué hacer a continuación

Elija cómo se autentican los usuarios:

Opciones de autenticación de IBM MQ Console

- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración del intervalo de caducidad de señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el certificado de cliente. Para obtener más información, consulte “Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console” en la página 511.

Opciones de autenticación de REST API

- Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte “Utilización de la autenticación básica HTTP con REST API” en la página 514.
- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. Para obtener más información, consulte “Utilización de la autenticación basada en señal con la API REST” en la página 516. Puede configurar el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración de la señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado de cliente. Para obtener más información, consulte “Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console” en la página 511.

ULW V 9.1.0 Configuración de un registro de SO local para la IBM MQ Console y la REST API

Puede configurar un registro de sistema operativo local en el archivo `mqwebuser.xml`. Los nombres de usuario y las contraseñas en el sistema operativo local se utilizan para autenticar y autorizar a los usuarios de la IBM MQ Console y la REST API.

Antes de empezar

- En la autenticación de certificados de cliente con la función de autenticación de sistema operativo local, la identidad de usuario es el nombre común (CN) del nombre distinguido (DN) del certificado de cliente. Si la identidad de usuario no existe como usuario del sistema operativo, el inicio de sesión de certificado de cliente fallará y recurrirá a una autenticación basada en contraseña.
- Para completar esta tarea, debe ser un usuario privilegiado.

Acerca de esta tarea

Con un registro del sistema operativo local, se asigna automáticamente un rol a los usuarios y los grupos.

- A cualquier usuario que forma parte del grupo 'mqm', o el grupo 'QMQMADM' en IBM i, se le otorgan los roles MQWebAdmin y MFTWebAdmin.
- A todos los demás usuarios se les otorga el rol MQWebUser.

Para obtener más información sobre estos roles, consulte "Roles en IBM MQ Console y REST API" en la página 509.

Solo se puede utilizar un registro de sistema operativo local en UNIX, Linux, and Windows. Se proporciona una función equivalente en z/OS configurando un registro SAF. Para obtener más información, consulte "Configuración de un registro SAF para la IBM MQ Console y REST API" en la página 506.

Procedimiento

1. Copie el archivo XML de ejemplo `local_os_registry.xml` de la vía de acceso siguiente:
   MQ_INSTALLATION_PATH`/web/mq/samp/configuration`
2. Coloque el archivo de ejemplo en el directorio siguiente:
   MQ_DATA_PATH`/web/installations/`*installationName*`/servers/mqweb`
3. Opcional: Si ha cambiado valores de configuración en `mqwebuser.xml`, cópielos en el archivo de ejemplo.
4. Suprima el archivo `mqwebuser.xml` existente y cambie el nombre del archivo de ejemplo a `mqwebuser.xml`.

Qué hacer a continuación

Elija cómo se autentican los usuarios:

Opciones de autenticación de IBM MQ Console

- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración del intervalo de caducidad de señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

Opciones de autenticación de REST API

- Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte "Utilización de la autenticación básica HTTP con REST API" en la página 514.
- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. Para obtener más información, consulte "Utilización de la autenticación basada en señal con la API REST" en la página 516. Puede configurar el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración de la señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado

de cliente. Para obtener más información, consulte “Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console” en la página 511.

V 9.1.0 Configuración de un registro LDAP para la IBM MQ Console y la REST API

Puede configurar un registro LDAP dentro del archivo `mqwebuser.xml`. Los nombres de usuario y las contraseñas del registro LDAP se utilizan para autenticar y autorizar usuarios de la IBM MQ Console y la REST API.

Antes de empezar

- Al configurar un registro LDAP, debe asignar a cada usuario un rol. Cada rol proporciona distintos niveles de privilegio para acceder a IBM MQ Console y REST API, y determina el contexto de seguridad que se utiliza cuando se intenta una operación permitida. Tendrá que comprender estos roles antes de configurar el registro. Si desea más información sobre cada uno de los roles, consulte “Roles en IBM MQ Console y REST API” en la página 509.

  Tenga en cuenta que cualquier usuario con el rol `MQWebUser` solo puede realizar las operaciones otorgadas al ID de usuario en el gestor de colas. Por lo tanto, el ID de usuario definido en el servidor LDAP debe tener un ID de usuario idéntico en el sistema en el cual está instalado IBM MQ. Estos ID de usuario tienen que coincidir en mayúsculas y minúsculas o la correlación entre ellos puede fallar.
- Para completar esta tarea, debe ser un usuario con privilegios suficientes para editar el archivo `mqwebuser.xml`:
  - z/OS En z/OS, debe tener acceso de escritura en el archivo `mqwebuser.xml`.
  - Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Procedimiento

1. Copie el archivo XML de ejemplo `ldap_registry.xml` de una de las vías de acceso siguientes:
   - ULW En UNIX, Linux, and Windows: *MQ_INSTALLATION_PATH* `/web/mq/samp/configuration`
   - z/OS En z/OS: `PathPrefix /web/mq/samp/configuration`

     donde `PathPrefix` es la vía de acceso de instalación de IBM MQ Unix System Services Components.
2. Coloque el archivo de ejemplo en el directorio adecuado:
   - ULW

     en UNIX, Linux, and Windows: *MQ_DATA_PATH*`/web/installations/`*installationName*`/servers/mqweb`
   - z/OS

     en z/OS: *WLP_user_directory*`/servers/mqweb`

     donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
3. Opcional: Si ha cambiado valores de configuración en `mqwebuser.xml`, cópielos en el archivo de ejemplo.
4. Suprima el archivo `mqwebuser.xml` existente y cambie el nombre del archivo de ejemplo a `mqwebuser.xml`.
5. Edite el nuevo archivo `mqwebuser.xml` para cambiar los valores del registro LDAP en los códigos **ldapRegistry** y **idsLdapFilterProperties**.

Para obtener más información sobre cómo configurar registros LDAP, consulte Configuración de registros de usuarios LDAP en Liberty en la documentación de WebSphere Liberty.

6. Asigne roles a usuarios y grupos editando el archivo `mqwebuser.xml`:

   Existen varios roles disponibles que autorizan a usuarios y grupos a utilizar IBM MQ Console, y REST API. Cada rol otorga un nivel de acceso diferente. Para obtener más información, consulte "Roles en IBM MQ Console y REST API" en la página 509.

   - Para asignar roles y otorgar acceso a IBM MQ Console, añada los usuarios y los grupos entre las etiquetas **`security-role`** adecuadas en las etiquetas **`<enterpriseApplication id="com.ibm.mq.console">`**.
   - Para asignar roles y otorgar acceso a REST API, añada los usuarios y los grupos entre las etiquetas **`security-role`** adecuadas en las etiquetas **`<enterpriseApplication id="com.ibm.mq.rest">`**.

Qué hacer a continuación

Elija cómo se autentican los usuarios:

Opciones de autenticación de IBM MQ Console

- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración del intervalo de caducidad de señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

Opciones de autenticación de REST API

- Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte "Utilización de la autenticación básica HTTP con REST API" en la página 514.
- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. Para obtener más información, consulte "Utilización de la autenticación basada en señal con la API REST" en la página 516. Puede configurar el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración de la señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado de cliente. Para obtener más información, consulte "Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console" en la página 511.

z/OS V 9.1.0 Configuración de un registro SAF para la IBM MQ Console y REST API

La interfaz de System Authorization Facility (SAF) permite al servidor mqweb llamar al gestor de seguridad externa para comprobar la autenticación y la autorización. A continuación, un usuario puede iniciar sesión en la IBM MQ Console y la REST API con un ID de usuario y una contraseña de z/OS.

Antes de empezar

- Al configurar un registro SAF, debe asignar usuarios a un rol. Cada rol proporciona distintos niveles de privilegio para acceder a IBM MQ Console y REST API, y determina el contexto de seguridad que se utiliza cuando se intenta una operación permitida. Tendrá que comprender estos roles antes de configurar el registro. Si desea más información sobre cada uno de los roles, consulte "Roles en IBM MQ Console y REST API" en la página 509.
- El proceso ángel de WebSphere Liberty debe estar ejecutándose para utilizar la interfaz autorizada con SAF. Consulte Habilitación de servicios autorizados de z/OS en Liberty para z/OS para obtener más información.
- Para completar esta tarea, debe tener acceso de escritura en el archivo `mqwebuser.xml`, y la autorización para definir perfiles de gestor de seguridad.

Nota: V9.1.0.20 A partir de IBM MQ 9.1.0 Fix Pack 20, el archivo de configuración de ejemplo `zos_saf_registry.xml` se ha actualizado para eliminar una entrada safAuthorization duplicada.

Esta actualización soluciona un problema en el que se puede producir un error ICH408I cuando el MQ Console en z/OS se actualiza a un nivel que envía WebSphere Liberty Profile 22.0.0.12 o posterior: es decir, desde IBM MQ 9.1.0 Fix Pack 15. Tener más de una sentencia safAuthorization no está soportado y puede provocar un error ICH408I cuando los usuarios que no están en los roles MQWebAdmin o MQWebAdminRO, en la clase EBJROLE, intentan acceder a un gestor de colas z/OS a través de MQ Console.

El valor predeterminado para **racRouteLog**, que especifica los tipos de intentos de acceso para registrar, es NONE. Si necesita un informe o registro adicional para la auditoría de seguridad, consulte SAF Authorization (safAuthorization) para obtener más información.

Acerca de esta tarea

La interfaz SAF permite al servidor mqweb llamar al gestor de seguridad externa para la autenticación y la comprobación de la autorización para ambas, IBM MQ Console y REST API.

Procedimiento

1. Siga los pasos en Habilitación de servicios autorizados z/OS en Liberty for z/OS para proporcionar acceso al servidor mqweb para utilizar servicios autorizados de z/OS.

   El JCL de ejemplo para iniciar el proceso ángel se encuentra en `USS_ROOT/web/templates/zos/procs/bbgzangl.jcl`, donde USS_ROOT es la vía de acceso en Unix System Services donde están instalados los componentes USS de IBM MQ for z/OS .

   En `bbgzangl.jcl`, cambie la sentencia SET ROOT para que apunte a `USS_ROOT/web`, por ejemplo, `/usr/lpp/mqm/V9R1M0/web`.

   Consulte Administración de Liberty en z/OS para obtener más información sobre cómo detener e iniciar el proceso ángel.
2. Siga los pasos en Liberty: Configuración del usuario sin autenticar de SAF (System Authorization Facility) para crear el usuario sin autenticar necesario para Liberty.
3. Copie el archivo `zos_saf_registry.xml` de la vía de acceso siguiente: `PathPrefix /web/mq/samp/configuration` donde `PathPrefix` es la vía de instalación de IBM MQ Unix System Services Components.
4. Coloque el archivo de ejemplo en el directorio *`WLP_user_directory`*`/servers/mqweb` , donde *WLP_user_directory* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
5. Opcional: Si ha cambiado anteriormente los valores de configuración en `mqwebuser.xml`, cópielos en el archivo de ejemplo.
6. Suprima el archivo `mqwebuser.xml` existente y cambie el nombre del archivo de ejemplo a `mqwebuser.xml`.
7. Personalice el elemento **safCredentials** en `mqwebuser.xml`.

a. Establezca **profilePrefix** en un nombre que sea exclusivo para el servidor Liberty. Si tiene más de un servidor mqweb en ejecución en un solo sistema, tendrá que elegir un nombre diferente para cada servidor; por ejemplo, MQWEB910 y MQWEB905.

b. Establezca **unauthenticatedUser** al nombre del usuario no autenticado creado en el paso "2" en la página 507.

8. Defina el APPLID del servidor mqweb en RACF.

   El nombre del recurso APPLID es el valor especificado en el atributo **profilePrefix** del paso "7" en la página 507. El ejemplo siguiente define el APPLID del servidor mqweb en RACF:

   ```
   RDEFINE APPL profilePrefix UACC(NONE)
   ```

9. Otorgue a todos los usuarios, o grupos, que tienen que autenticarse en MQ Console o REST API acceso READ al servidor mqweb APPLID en la clase APPL.

   También hay que hacer esto para el usuario no autenticado definido en el paso "2" en la página 507. El ejemplo siguiente otorga acceso READ de usuario al APPLID del servidor mqweb en RACF:

   ```
   PERMIT profilePrefix CLASS(APPL) ACCESS(READ) ID(userID)
   ```

10. Defina los perfiles en la clase EJBROLE necesarios para proporcionar a los usuarios acceso a los roles en MQ Console y REST API.

    El ejemplo siguiente define los perfiles en RACF, donde **profilePrefix** es el valor especificado para el atributo **profilePrefix** en el paso "7" en la página 507.

```
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.console.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebAdminRO UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MQWebUser UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdmin UACC(NONE)
RDEFINE EJBROLE profilePrefix.com.ibm.mq.rest.MFTWebAdminRO UACC(NONE)
```

11. Otorgue a los usuarios acceso a los roles en MQ Console y REST API.

    Para ello, proporcione a los usuarios o grupos acceso READ a uno o varios de los perfiles en la clase EBJROLE creada en el paso "10" en la página 508. Si desea más información sobre los roles, consulte "Roles en IBM MQ Console y REST API" en la página 509.

    El ejemplo siguiente proporciona acceso de usuario al rol MQWebAdmin para la REST API en RACF, donde **profilePrefix** es el valor especificado para el atributo **profilePrefix** en el paso "7" en la página 507.

```
PERMIT profilePrefix.com.ibm.mq.rest.MQWebAdmin CLASS(EJBROLE) ACCESS(READ) ID(userID)
```

Resultados

Ha configurado autenticación SAP para IBM MQ Console y REST API.

Qué hacer a continuación

Elija cómo se autentican los usuarios:

Opciones de autenticación de IBM MQ Console

- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario especifica un ID de usuario y una contraseña en la pantalla de inicio de sesión de IBM MQ Console. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. No se requiere ninguna configuración adicional para utilizar esta opción de autenticación, pero puede configurar de manera opcional el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración del intervalo de caducidad de señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la IBM MQ Console, sino que utiliza el

certificado de cliente. Para obtener más información, consulte “Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console” en la página 511.

Opciones de autenticación de REST API

- Los usuarios se autentican utilizando la autenticación básica HTTP. En este caso, se codifican un nombre de usuario y una contraseña, pero no se cifran, y se envían a cada solicitud REST API para autenticar y autorizar al usuario para esa solicitud. Para que esta autenticación sea segura, debe utilizar una conexión segura. Es decir, debe utilizar HTTPS. Para obtener más información, consulte “Utilización de la autenticación básica HTTP con REST API” en la página 514.
- Los usuarios se autentican utilizando la autenticación de señal. En este caso, un usuario proporciona un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario permanecer conectado y autorizado durante una cantidad de tiempo definida. Para obtener más información, consulte “Utilización de la autenticación basada en señal con la API REST” en la página 516. Puede configurar el intervalo de caducidad de la señal LTPA. Para obtener más información, consulte Configuración de la señal LTPA.
- Los usuarios se autentican utilizando certificados de cliente. En este caso, el usuario no utiliza un ID de usuario o una contraseña para iniciar una sesión en la REST API, sino que utiliza el certificado de cliente. Para obtener más información, consulte “Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console” en la página 511.

V 9.1.0 Roles en IBM MQ Console y REST API

Cuando autoriza a usuarios y grupos a utilizar la IBM MQ Console o REST API, debe asignar a los usuarios y grupos uno de los roles disponibles: **MQWebAdmin**, **MQWebAdminRO**, **MQWebUser**, **MFTWebAdmin** y **MFTWebAdminRO**. Cada rol proporciona distintos niveles de privilegio para acceder a IBM MQ Console y REST API, y determina el contexto de seguridad que se utiliza cuando se intenta una operación permitida.

Nota: Con la excepción del rol **MQWebUser** , el ID de usuario no distingue entre mayúsculas y minúsculas. Consulte “MQWebUser” en la página 510 para conocer los requisitos específicos de este rol.

MQWebAdmin

Un usuario o grupo que tiene asignado este rol puede realizar todas las operaciones administrativas y funciona bajo el contexto de seguridad del ID de usuario del sistema operativo que se ha utilizado para iniciar el servidor mqweb.

Un usuario o grupo con este rol no tiene acceso a los servicios REST siguientes:

- La REST API para MFT. Para utilizar estos servicios, el usuario o grupo también debe tener asignado el rol **MFTWebAdmin** o **MFTWebAdminRO**.
- messaging REST API. Para utilizar la messaging REST API, el usuario debe tener asignado el rol **MQWebUser**.

MQWebAdminRO

Este rol proporciona acceso de sólo lectura a la IBM MQ Console o la REST API. Un usuario o grupo al que se asigna este rol puede realizar las operaciones siguientes:

- Visualizar y consultar las operaciones en objetos de IBM MQ, como colas y canales.
- Examinar mensajes en colas.

Un usuario o grupo al que se asigna este rol opera bajo el contexto de seguridad del ID de usuario de sistema operativo que se utiliza para iniciar el servidor mqweb.

Un usuario o grupo con este rol no tiene acceso a los servicios REST siguientes:

- La REST API para MFT. Para utilizar estos servicios, el usuario o grupo también debe tener asignado el rol **MFTWebAdmin** o **MFTWebAdminRO**.
- messaging REST API. Para utilizar la messaging REST API, el usuario debe tener asignado el rol **MQWebUser**.

MQWebUser
Un usuario o grupo al que se asigne este rol podrá realizar cualquier operación permitida al ID de usuario en el gestor de colas. Por ejemplo:

- Iniciar y detener las operaciones en objetos de IBM MQ como canales.
- Definir y establecer operaciones en objetos de IBM MQ como colas y canales.
- Visualizar y consultar las operaciones en objetos de IBM MQ, como colas y canales.
- Coloque y obtenga mensajes utilizando la messaging REST API.

Un usuario o grupo al que se asigna este rol opera bajo el contexto de seguridad del principal y sólo puede realizar las operaciones que el ID de usuario está autorizado a realizar en el gestor de colas.

Por lo tanto, al usuario o al grupo que se define en el registro de usuarios mqweb debe otorgársele autorización en IBM MQ para que el usuario pueda realizar alguna operación. Mediante este rol, puede controlar con precisión qué usuarios tienen qué tipo de acceso a recursos específicos de IBM MQ cuando utilizan IBM MQ Console y REST API.

Nota:

- La longitud máxima de un ID de usuario al que se asigna este rol es de 12 caracteres.
- El ID de usuario tiene que coincidir en mayúsculas y minúsculas con del registro de usuarios de mqweb y en el sistema IBM MQ. Si las mayúsculas y minúsculas del ID de usuario son diferentes, el usuario podría ser autenticado por la IBM MQ Console y la REST API, pero no estar autorizado para utilizar recursos de IBM MQ.

Un usuario o grupo con este rol no tiene acceso a ninguno de los servicios de la REST API para MFT. Para utilizar estos servicios, el usuario o grupo también debe tener asignado el rol **MFTWebAdmin** o **MFTWebAdminRO**.

MFTWebAdmin
Un usuario o grupo asignado a este rol puede realizar todas las operaciones REST de MFT y opera bajo el contexto de seguridad del ID de usuario del sistema operativo que se utiliza para iniciar el servidor `mqweb`.

Un usuario o grupo con este rol no tiene acceso a ninguno de los servicios de IBM MQ REST API. Para utilizar estos servicios, el usuario o grupo también debe tener asignado el rol **MQWebAdmin**, **MQWebAdminRO** o **MQWebUser**.

MFTWebAdminRO
Este rol proporciona acceso de sólo lectura a la REST API para MFT . Un usuario o grupo que tiene asignado este rol puede realizar operaciones de sólo lectura (solicitudes GET) como listar transferencia y listar agentes.

Un usuario o grupo al que se asigna este rol opera bajo el contexto de seguridad del ID de usuario de sistema operativo que se utiliza para iniciar el servidor mqweb.

Un usuario o grupo con este rol no tiene acceso a ninguno de los servicios de IBM MQ REST API. Para utilizar estos servicios, el usuario o grupo también debe tener asignado el rol **MQWebAdmin**, **MQWebAdminRO** o **MQWebUser**.

Para obtener más información sobre cómo configurar usuarios y grupos para utilizar estos roles, consulte “Configuración de usuarios y roles” en la página 499.

Solapamiento de roles

A un usuario o grupo se le puede asignar más de un rol. Cuando un usuario realiza una operación en esta situación, se utiliza el rol de mayor privilegio que es aplicable a la operación. Por ejemplo, si un usuario con los roles **MQWebAdminRO** y **MQWebUser** realiza una operación de cola de consulta, se utiliza el rol **MQWebAdminRO** y se intenta la operación bajo el contexto del ID de usuario de sistema que ha iniciado el servidor web. Si dicho mismo usuario realiza una operación de definición, se utiliza el rol **MQWebUser** y se intenta la operación bajo el contexto del principal.

ULW V 9.1.0 Utilización de la autenticación de certificado de cliente con REST API y IBM MQ Console

Puede correlacionar certificados de cliente con principales para autenticar usuarios de IBM MQ Console y REST API.

Antes de empezar

- Configure los usuarios, los grupos y los roles a los que se les va a autorizar el uso de IBM MQ Console y REST API. Para obtener más información, consulte "Configuración de usuarios y roles" en la página 499.
- Cuando utiliza REST API, puede consultar las credenciales del usuario actual utilizando el método HTTP GET en el recurso `login`, proporcionando el certificado de cliente para autenticar la solicitud. Esta solicitud devuelve información sobre el nombre de usuario, y los roles a los que está asignado el usuario. Para obtener más información, consulte GET `/login`.
- Cuando los certificados de cliente se correlacionan con principales para autenticar usuarios, se utiliza el nombre distinguido del certificado de cliente para compararlo con los usuarios del registro de usuarios configurado:
  - Para un registro básico, se compara el nombre común (CN) con el usuario. Por ejemplo, `CN=Fred, O=IBM, C=GB` se compara con un nombre de usuario de `Fred`.
  - Para un registro LDAP, el nombre distinguido completo se compara con LDAP de forma predeterminada. Puede configurar filtros y correlación para personalizar la coincidencia. Para obtener más información, consulte Liberty :modalidad de correlación de certificados LDAP en la documentación de WebSphere Liberty .

Acerca de esta tarea

Cuando un usuario se autentica utilizando un certificado de cliente, se utiliza el certificado en lugar de un nombre de usuario y una contraseña. Para REST API, el certificado de cliente se proporciona con cada solicitud REST para autenticar el usuario. Para IBM MQ Console, cuando un usuario inicia una sesión con un certificado, no puede cerrarla.

En el procedimiento se supone la siguiente información:

- Que el archivo `mqwebuser.xml` se basa en uno de los ejemplos siguientes:
  - `basic_registry.xml`
  - `local_os_registry.xml`
  - `ldap_registry.xml`
- Que está utilizando un sistema UNIX, Linux o Windows.
- Que es un usuario privilegiado.

Para configurar la autenticación de certificado de cliente con un conjunto de claves RACF en z/OS, siga el procedimiento en "Configuración de TLS para REST API y IBM MQ Console en z/OS" en la página 523.

Nota: En el siguiente procedimiento se describen los pasos necesarios para utilizar certificados de cliente con IBM MQ Console y REST API. Para la comodidad del desarrollador, los pasos detallan cómo crear y utilizar certificados autofirmados. No obstante, para la producción, utilice certificados obtenidos de una entidad emisora de certificados.

Procedimiento

1. Inicie el servidor mqweb especificando el mandato **strmqweb** en la línea de mandatos.
2. Cree un certificado de cliente:
   a) Cree un almacén de claves de PKCS#12:
      i) Abra la herramienta IBM Key Management especificando el mandato **strmqikm** en la línea de mandatos.

ii) En el menú **Archivo de base de datos de claves** en la herramienta IBM Key Management, pulse **Nuevo**.

iii) Seleccione **PKCS12** en la lista **Tipo de base de datos de claves**.

iv) Seleccione una ubicación para guardar el almacén de claves y especifique un nombre adecuado en el campo **Nombre de archivo**. Por ejemplo, `user.p12`

v) Establezca una contraseña cuando se le solicite.

b) Cree el certificado, ya sea creando un certificado autofirmado u obteniendo un certificado de una autoridad certificadora:

- Crear un certificado autofirmado:

  i) Pulse **Nuevo autofirmado**.

  ii) Especifique `user` en el campo **Etiqueta de clave**.

  iii) Si utiliza un registro de usuarios básico, especifique el nombre de un usuario del registro de usuarios en el campo **Nombre común**. Por ejemplo, `mqadmin`. Para un registro de usuarios LDAP, asegúrese de que el nombre distinguido del certificado coincida con el nombre distinguido en el registro LDAP.

  iv) Pulse **Aceptar**.

- Obtenga un certificado de una entidad emisora de certificados. El certificado de CA debe incluir el nombre de usuario apropiado dentro del nombre común (CN) del campo de nombre distinguido (DN):

  i) Solicite un nuevo certificado. En el menú **Crear**, pulse **Nueva solicitud de certificado**.

  ii) En el campo **Etiqueta de clave**, escriba la etiqueta del certificado.

  iii) Si utiliza un registro de usuarios básico, en el campo **Nombre común**, especifique el nombre del usuario para el que se proporciona el certificado.

  Si está utilizando un registro del sistema operativo local, el campo **Nombre común** tiene que coincidir con el ID de usuario del sistema operativo local.

  Para un registro de usuarios LDAP, asegúrese de que el nombre distinguido del certificado coincida con el nombre distinguido en el registro LDAP.

  iv) Especifique o seleccione valores para los campos restantes, según corresponda.

  v) Elija dónde desea guardar la solicitud de certificado y el nombre de archivo de la solicitud de certificado y, a continuación, pulse **Aceptar**.

  vi) Envíe el archivo de solicitud de certificado a una entidad emisora de certificados (CA).

  vii) Cuando tenga el certificado de la CA, abra la herramienta IBM Key Management especificando el mandato **`strmqikm`** en la línea de mandatos.

  viii) En el menú **Archivo de base de datos de claves** en la herramienta IBM Key Management, pulse **Abrir**.

  ix) Seleccione el almacén de claves PKCS#12 que contiene el certificado de cliente. Por ejemplo: `user.p12`

  x) Pulse **Recibir**, seleccione el certificado adecuado y pulse **Aceptar**.

3. Extraiga la parte pública del certificado de cliente:

a) Abra la herramienta IBM Key Management especificando el mandato **`strmqikm`** en la línea de mandatos.

b) En el menú **Archivo de base de datos de claves** en la herramienta IBM Key Management, pulse **Abrir**.

c) Seleccione el almacén de claves PKCS#12 que contiene el certificado de cliente. Por ejemplo: `user.p12`

d) Seleccione el certificado de cliente en la lista de certificados en la herramienta IBM Key Management.

e) Pulse **Extraer certificado**.

f) Seleccione una ubicación donde guardar el certificado y especifique un nombre de archivo adecuado en el campo **Nombre de archivo de certificado**. Por ejemplo, `user.arm`.

4. Importe la parte pública del certificado de cliente al almacén de claves de confianza del servidor mqweb como un certificado de firmante, para que el servidor pueda validar el certificado de cliente:

a) Si no existe, cree un almacén de claves `trust.jks` para que lo use el servidor mqweb:

i) En el menú **Archivo de base de datos de claves** en la herramienta IBM Key Management, pulse **Nuevo**.

ii) Seleccione **JKS** en la lista **Tipo de base de datos de claves**.

iii) Pulse **Examinar** y vaya a: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb/resources/security`.

Este directorio ya debería contener un archivo `key.jks`. Si ya existe un archivo `trust.jks`, ábralo en lugar de sobrescribirlo.

iv) Especifique `trust.jks` en el campo **Nombre de archivo**.

v) Establezca una contraseña cuando se le solicite.

b) Desde el menú desplegable, seleccione **Certificados de firmante**.

c) Pulse **Añadir**.

d) Seleccione el archivo de brazo adecuado y pulse **Aceptar**. Por ejemplo, seleccione `user.arm`.

e) Especifique una etiqueta para el certificado.

5. Cambie la contraseña del almacén de claves del servidor mqweb:

a) En el menú **Archivo de base de datos de claves**, pulse **Abrir**.

b) Seleccione **JKS** en la lista **Tipo de base de datos de claves**.

c) Pulse **Examinar** y vaya a *MQ_DATA_PATH*/web/installations/*installationName*/servers/mqweb/resources/security

d) Seleccione el almacén de claves `key.jks` y pulse **Abrir**.

e) Especifique la contraseña cuando se le solicite. La contraseña predeterminada es `password`.

f) En el menú **Archivo de base de datos de claves**, pulse **Cambiar contraseña**.

g) Especifique una nueva contraseña para el almacén de claves.

6. Habilite la autenticación de certificados de cliente en el archivo `mqwebuser.xml`:

El archivo `mqwebuser.xml` se puede encontrar en la siguiente vía de acceso: *MQ_DATA_PATH*/web/installations/*installationName*/servers/mqweb

a) Elimine el comentario de la sección del archivo `mqwebuser.xml` que permite la autenticación de certificados de cliente. La sección contiene el texto siguiente:

```
<keyStore id="defaultKeyStore" location="key.jks" type="JKS" password="password"/>
    <keyStore id="defaultTrustStore" location="trust.jks" type="JKS" password="password"/>
    <ssl id="thisSSLConfig" clientAuthenticationSupported="true"
keyStoreRef="defaultKeyStore"
          trustStoreRef="defaultTrustStore" sslProtocol="TLSv1.2"
serverKeyAlias="default"/>
    <sslDefault sslRef="thisSSLConfig"/>
```

b) Compruebe que el valor **serverKeyAlias** coincida con el nombre del certificado de servidor. Si utiliza el certificado de servidor predeterminado, el valor es correcto.

c) Cambie el valor de **password** en `defaultKeyStore` por una versión codificada de la contraseña para el almacén de claves `key.jks`:

i) En el directorio *MQ_INSTALLATION_PATH*/web/bin , especifique el mandato siguiente en la línea de mandatos:

```
securityUtility encode password
```

ii) Coloque la salida de este mandato en el campo **password** de `defaultKeyStore`.

d) Cambie el valor de **password** en `defaultTrustStore` para que coincida con la contraseña del almacén de claves `trust.jks`:

i) En el directorio *MQ_INSTALLATION_PATH*`/web/bin` , especifique el mandato siguiente en la línea de mandatos:

```
securityUtility encode password
```

ii) Coloque la salida de este mandato en el campo **password** de `defaultTrustStore`.

e) Elimine, o elimine el comentario de, la línea siguiente del archivo `mqwebuser.xml`:

```
<sslDefault sslRef="mqDefaultSSLConfig"/>
```

7. Detenga el servidor mqweb especificando el mandato **endmqweb** en la línea de mandatos.
8. Inicie el servidor mqweb especificando el mandato **strmqweb** en la línea de mandatos.
9. Utilice el certificado de cliente para la autenticación:
   - Para utilizar el certificado de cliente con IBM MQ Console, instale el certificado de cliente en el navegador web que se utiliza para acceder a IBM MQ Console. Por ejemplo, instale el certificado de cliente `user.p12` como un certificado personal.
   - Para utilizar el certificado de cliente con REST API, proporcione el certificado de cliente con cada solicitud REST. Cuando utiliza los métodos HTTP POST, PATCH o DELETE, debe proporcionar una autenticación adicional con el certificado de cliente para evitar ataques de falsificación de solicitudes entre sitios. Es decir, la autenticación adicional se utiliza para confirmar que las credenciales para autenticar la solicitud las utiliza el propietario de las credenciales.

     Esta autenticación adicional la proporciona la cabecera HTTP `ibm-mq-rest-csrf-token`. Establezca el valor de la cabecera `ibm-mq-csrf-token` en cualquier valor, incluido el espacio en blanco y, después, envíe la solicitud.

Ejemplo

Importante: En el ejemplo, no todas las implementaciones de cURL admiten los certificados firmados automáticamente, así que debe utilizar una implementación de cURL que lo haga.

El siguiente ejemplo de cURL muestra cómo crear una nueva cola `Q1` en el gestor de colas `QM1` con autenticación de certificado de cliente. La configuración exacta de este mandato cURL depende de las bibliotecas con las que se ha creado cURL. El ejemplo se basa en un sistema Windows, con el cURL creado con respecto a OpenSSL.

- Utilice el método HTTP POST con el recurso de cola y autentíquese con el certificado de cliente, incluyendo el contenido de la cabecera HTTP `ibm-mq-rest-csrf-token` con un valor arbitrario. Este valor puede ser cualquier valor, incluido el espacio en blanco. El distintivo `--cert-type` especifica que el certificado es un certificado PKCS#12. El distintivo `--cert` especifica la ubicación del certificado, seguida por dos puntos, : y, después, la contraseña para el certificado.

```
curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -
-cert-type P12 --cert c:\user.p12:password
-H "ibm-mq-rest-csrf-token: value"
-H "Content-Type: application/json" --data "{\"name\":\"Q1\"}"
```

V 9.1.0 Utilización de la autenticación básica HTTP con REST API

Los usuarios de REST API pueden autenticarse proporcionando su ID de usuario y su contraseña en una cabecera HTTP. Para utilizar este método de autenticación con métodos HTTP como, por ejemplo, POST, PATCH y DELETE, la cabecera HTTP `ibm-mq-rest-csrf-token` también se debe proporcionar, así como un ID de usuario y una contraseña.

Antes de empezar

- Configure los usuarios, grupos y roles a los que se les va a autorizar a utilizar REST API. Para obtener más información, consulte “Configuración de usuarios y roles” en la página 499.
- Asegúrese de que la autenticación básica HTTP esté habilitada. Compruebe que el siguiente XML esté presente y no comentado en el archivo `mqwebuser.xml`. El XML debe estar dentro de las etiquetas `<featureManager>`:

  ```
  <feature>basicAuthenticationMQ-1.0</feature>
  ```

  z/OS En z/OS, debe ser un usuario que tenga acceso de escritura a `mqwebuser.xml` para editar este archivo.

  Multi En todos los demás sistemas operativos, debe ser un usuario privilegiado para editar el archivo `mqwebuser.xml`.
- Asegúrese de que esté utilizando una conexión segura cuando envíe solicitudes REST. Como la combinación de nombre de usuario y contraseña está codificada, pero no cifrada, debe utilizar una conexión segura (HTTPS) cuando se utiliza la autenticación básica HTTP con REST API.
- Para consultar las credenciales del usuario actual, utilice el método HTTP GET en el recurso `login` y proporcione la información de autenticación básica para autenticar la solicitud. Esta solicitud devuelve información sobre el nombre de usuario, y los roles a los que está asignado el usuario. Para obtener más información, consulte GET `/login`.

Procedimiento

1. Concatene el nombre de usuario con dos puntos y la contraseña. Tenga en cuenta que el nombre de usuario distingue entre mayúsculas y minúsculas.

   Por ejemplo, un nombre de usuario admin y una contraseña admin se convierten en la siguiente serie:

   ```
   admin:admin
   ```
2. Codifique esta serie de nombre de usuario y contraseña en codificación base64.
3. Incluya este nombre de usuario y contraseña codificados en una cabecera HTTP `Authorization: Basic`.

   Por ejemplo, con un nombre de usuario codificado admin y una contraseña admin, se crea la siguiente cabecera:

   ```
   Authorization: Basic YWRtaW46YWRtaW4=
   ```
4. Cuando utiliza los métodos HTTP POST, PATCH o DELETE, debe proporcionar una autenticación adicional, así como un nombre de usuario y una contraseña.

   Esta autenticación adicional la proporciona la cabecera HTTP `ibm-mq-rest-csrf-token`. La cabecera HTTP `ibm-mq-rest-csrf-token` debe estar presente en la solicitud, pero su valor puede ser cualquier valor, incluyendo espacios en blanco.
5. Envíe la solicitud REST a IBM MQ con las cabeceras correspondientes.

Ejemplo

El siguiente ejemplo muestra cómo crear una nueva cola Q1 en el gestor de colas QM1, con la autenticación básica, en sistemas Windows. El ejemplo utiliza el cURL:

- Utilice el método HTTP POST con el recurso de cola y autentíquese con la autenticación básica, incluyendo la cabecera HTTP `ibm-mq-rest-csrf-token` con un valor arbitrario. Este valor puede ser cualquier valor, incluyendo los espacios en blanco:

  ```
  curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST
  -u mqadmin:mqadmin
  ```

```
-H "ibm-mq-rest-csrf-token: value"
-H "Content-Type: application/json" --data "{\"name\":\"Q1\"}"
```

V 9.1.0 Utilización de la autenticación basada en señal con la API REST

Los usuarios de REST API pueden autenticarse proporcionando un ID de usuario y una contraseña al recurso REST API `login` con el método HTTP POST. Se genera una señal LTPA que permite al usuario autenticar solicitudes en el futuro. Esta señal LTPA tiene el prefijo `LtpaToken2`. El usuario puede finalizar la sesión utilizando el método HTTP DELETE y puede consultar la información de inicio de sesión del usuario actual con el método HTTP GET.

Antes de empezar

- Configure los usuarios, grupos y roles a los que se les va a autorizar a utilizar REST API. Para obtener más información, consulte “Configuración de usuarios y roles” en la página 499.
- De forma predeterminada, el nombre de la cookie que incluye la señal LTPA empieza con `LtpaToken2` e incluye un sufijo que puede cambiar cuando se reinicia el servidor mqweb. Este nombre de cookie aleatorizado permite que se pueda ejecutar más de un servidor mqweb en el mismo sistema. Sin embargo, si desea que el nombre de la cookie siga siendo un valor coherente, puede especificar el nombre que tiene la cookie utilizando el mandato **setmqweb**. Para obtener más información, consulte Configuración de la señal LTPA.
- De forma predeterminada, la cookie de la señal LTPA caduca después de 120 minutos. Puede configurar la hora de caducidad de la cookie de la señal LTPA utilizando el mandato **setmqweb**. Para obtener más información, consulte Configuración de la señal LTPA.
- Asegúrese de que esté utilizando una conexión segura cuando envíe solicitudes REST. Cuando utiliza el método HTTP POST en el recurso `login`, la combinación de nombre de usuario y contraseña que se envía con la solicitud no están cifrados. Por lo tanto, debe utilizar una conexión segura (HTTPS) cuando utiliza la autenticación basada en señal con REST API. De forma predeterminada, no puede utilizar HTTP con la autenticacion de señal LTPA. Puede habilitar la señal LTPA para que sea utilizada por conexiones HTTP no seguras estableciendo **secureLTPA** en `False`. Para obtener más información, consulte Configuración de la señal LTPA.
- Puede consultar las credenciales del usuario actual utilizando el método HTTP GET en el recurso `login`, proporcionando la señal LTPA para autenticar la solicitud. Esta solicitud devuelve información sobre el nombre de usuario, y los roles a los que está asignado el usuario. Para obtener más información, consulte GET `/login`.

Procedimiento

1. Inicie una sesión de un usuario:
   a) Utilice el método HTTP POST en el recurso `login`:

   ```
   https://host:port/ibmmq/rest/v1/login
   ```

   Incluya el nombre de usuario y la contraseña en el cuerpo de la solicitud JSON, con el formato siguiente:

   ```
   {
       "username" : name,
       "password" : password
   }
   ```

   b) Almacene la señal LTPA que se ha devuelto de la solicitud en el almacén de cookies local. De forma predeterminada, esta señal LTPA tiene un prefijo de `LtpaToken2`.
2. Autentique las solicitudes REST con la señal LTPA almacenada como una cookie con cada solicitud.

   Para las solicitudes que utilizan los métodos HTTP PUT, PATCH o DELETE, incluya una cabecera `ibm-mq-rest-csrf-token`. El valor de esta cabecera puede ser cualquier elemento, incluso estar en blanco.

3. Cierre la sesión de un usuario:
   a) Utilice el método HTTP DELETE en el recurso `login`:

   ```
   https://host:9443/ibmmq/rest/v1/login
   ```

   Debe proporcionar la señal LTPA como una cookie para autenticar la solicitud e incluir una cabecera `ibm-mq-rest-csrf-token`. El valor de esta cabecera puede ser cualquier elemento, incluso estar en blanco.

   b) Procese la instrucción para suprimir la señal LTPA del almacén de cookies local.

   Nota: Si la instrucción no se procesa y la señal LTPA permanece en el almacén de cookies local, la señal LTPA puede utilizarse para autenticar solicitudes REST en el futuro. Es decir, cuando el usuario intenta autenticarse con la señal LTPA una vez finalizada la sesión, se crea una nueva sesión que utiliza la señal existente.

Ejemplo

El siguiente ejemplo de cURL muestra cómo crear una nueva cola Q1 en el gestor de colas QM1, con la autenticación basada en señal, en sistemas Windows:

- Inicie sesión y añada la señal LTPA con el prefijo `LtpaToken2` al almacén de cookies local. La información de nombre de usuario y contraseña se incluye en el cuerpo JSON. El distintivo `-c` especifica la ubicación del archivo donde se almacena la señal:

  ```
  curl -k https://localhost:9443/ibmmq/rest/v1/login -X POST
  -H "Content-Type: application/json" --data
  "{\"username\":\"mqadmin\",\"password\":\"mqadmin\"}"
  -c c:\cookiejar.txt
  ```

- Cree una cola. Utilice el método HTTP POST con el recurso de cola y autentíquese con la señal LTPA. La señal LTPA con el prefijo `LtpaToken2` se recupera del archivo `cookiejar.txt` utilizando el código `-b`. La presencia de la cabecera HTTP `ibm-mq-rest-csrf-token` proporcionan protección CSRF:

  ```
  curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X POST -b
  c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: value" -H "Content-Type: application/json"
  --data "{\"name\":\"Q1\"}"
  ```

- Cierre la sesión y suprima la señal LTPA del almacén de cookies local. La señal LTPA se recupera del archivo `cookiejar.txt` utilizando el código `-b`. La presencia de la cabecera HTTP `ibm-mq-rest-csrf-token` proporciona protección CSRF. La ubicación del archivo `cookiejar.txt` se especifica mediante el distintivo `-c` para que la señal LTPA se suprima del archivo:

  ```
  curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1/queue -X DELETE
  -H "ibm-mq-rest-csrf-token: value" -b c:\cookiejar.txt
  -c c:\cookiejar.txt
  ```

Referencia relacionada

POST/login
GET/login
Suprimir /login

V 9.1.3 Inclusión de la IBM MQ Console en un cuadro de información

Se puede utilizar el elemento HTML <iframe> para incluir una página web en otra utilizando un marco flotante (IFrame). Por razones de seguridad, la IBM MQ Console no puede estar incluida en un IFrame de forma predeterminada. Sin embargo, puede habilitar un IFrame utilizando la propiedad de configuración **mqConsoleFrameAncestors** en el servidor mqweb.

Acerca de esta tarea

El servidor mqweb mantiene una lista de elementos permitidos de orígenes de páginas web que pueden incluir IBM MQ Console utilizando un IFrame. Un origen es una combinación de un esquema, dominio y puerto de URL, por ejemplo, `https://example.com:1234`.

Puede utilizar la propiedad de configuración **mqConsoleFrameAncestors** en el servidor mqweb para especificar las entradas de la lista.

De forma predeterminada, **mqConsoleFrameAncestors** está en blanco, lo que significa que la IBM MQ Console no se puede incorporar en un IFrame.

Procedimiento

Especifique una lista de orígenes de páginas web, que puede incluir la IBM MQ Console en un IFrame, especificando el mandato siguiente:

```
setmqweb properties -k mqConsoleFrameAncestors -v allowedOrigins
```

donde *allowedOrigins* es una lista separada por comas de los orígenes. Cada origen debe estar formado por:

- Un nombre de host o una dirección IP
- Un esquema de URL opcional
- Un número de puerto opcional

Tenga en cuenta que el nombre de host puede empezar con el carácter comodín (*) y que el número de puerto también puede utilizar el carácter comodín (*).

Los orígenes de ejemplo son:

```
https://example.com:1234
```

que permite que cualquier página web servida desde `https://example.com:1234` incluya IBM MQ Console en un IFrame.

```
https://*.example.com:*
```

que permite que cualquier página web HTTPS con un nombre de host que termine en `example.com`, y que utilice cualquier puerto, incluya IBM MQ Console en un IFrame.

Ejemplo

El ejemplo siguiente permite que la inclusión de la IBM MQ Console en un IFrame desde páginas web proporcionadas desde `https://site2.example.com:1234` o `https://site2.example.com:1235`:

```
setmqweb properties -k mqConsoleFrameAncestors -v
https://site2.example.com:1234,https://site2.example.com:1235
```

V 9.1.0 Configuración de CORS para REST API

De forma predeterminada, un navegador web no permite que scripts como, por ejemplo, JavaScript, invoquen REST API cuando no provienen del mismo origen que REST API. Es decir, las solicitudes entre orígenes no están habilitadas. Puede configurar Cross Origin Resource Sharing (CORS) para permitir las solicitudes entre orígenes a partir de los orígenes especificados.

Acerca de esta tarea

Puede acceder a REST API mediante un navegador web, por ejemplo, mediante un script. Como estas solicitudes son de un origen diferente a REST API, el navegador web rechaza la solicitud porque es una solicitud entre orígenes. El origen es diferente si el dominio, el puerto o el esquema no son los mismos.

Por ejemplo, si tiene un script que se aloja en `http://localhost:1999/`, realiza una solicitud entre orígenes si emite un HTTP GET en un sitio web que está alojado en `https://localhost:9443/`. Esta solicitud es una solicitud entre orígenes porque los números de puerto y el esquema (HTTP) son diferentes.

Para habilitar solicitudes entre orígenes, configure CORS y especifique los orígenes que pueden acceder REST API.

Para obtener más información sobre CORS, consulte https://www.w3.org/TR/cors/ y https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS.

Procedimiento

1. Consulte la configuración actual especificando el mandato siguiente:

   ```
   dspmqweb properties -a
   ```

   La entrada `mqRestCorsAllowedOrigins` especifica los orígenes permitidos. La entrada `mqRestCorsMaxAgeInSeconds` especifica el tiempo, en segundos, durante el cual el navegador web puede almacenar en memoria caché los resultados de cualquier comprobación previa al lanzamiento de CORS.
2. Especifique los orígenes que pueden acceder a REST API mediante el mandato siguiente:

   ```
   setmqweb properties -k mqRestCorsAllowedOrigins -v allowedOrigins
   ```

   donde *allowedOrigins* especifica el origen desde el que se desea permitir solicitudes de orígenes. Puede utilizar un asterisco encerrado entre comillas dobles ("*") para permitir todas las peticiones entre orígenes. Puede especificar más de un origen en una lista separada por comas, encerrados entre comillas dobles. Para no permitir solicitudes entre orígenes, especifique comillas dobles vacías como valor de *allowedOrigins*.
3. Especifique el tiempo, en segundos, que desea permitir que un navegador web almacene en caché el resultado de las comprobaciones preparatorias CORS ejecutando el mandato siguiente:

   ```
   setmqweb properties -k mqRestCorsMaxAgeInSeconds -v time
   ```

Ejemplo

El siguiente ejemplo muestra las solicitudes entre orígenes habilitadas para `http://localhost:9883`, `https://localhost:1999` y `https://localhost:9663`. La antigüedad máxima de los resultados en la memoria caché de cualquier comprobación previa de CORS se establece en 90 segundos:

```
setmqweb properties -k mqRestCorsAllowedOrigins -v "http://localhost:9883,https://
localhost:1999,https://localhost:9663"
setmqweb properties -k mqRestCorsMaxAgeInSeconds -v 90
```

Configurando la validación de la cabecera de host para la IBM MQ Console y la REST API

Puede configurar el servidor mqweb para restringir el acceso a IBM MQ Console y REST API de tal forma que sólo se procesen las solicitudes que se envían con una cabecera de host que coincida con una lista de elementos permitidos especificada. Se devuelve un error si se utiliza un valor de cabecera de host que no está en la lista de elementos permitidos.

Acerca de esta tarea

El servidor mqweb utiliza hosts virtuales para definir la lista de elementos permitidos de cabeceras de host aceptables. Para obtener más información sobre los hosts virtuales, consulte la documentación de WebSphere Liberty : https://www.ibm.com/docs/SSEQTP_liberty/com.ibm.websphere.wlp.doc/ae/cwlp_virtual_hosts.html

Para completar esta tarea, debe ser un usuario con privilegios suficientes para editar el archivo `mqwebuser.xml`:

- z/OS En z/OS, debe tener acceso de escritura en el archivo `mqwebuser.xml`.
- Multi En todos los demás sistemas operativos, debe ser un usuario con privilegios.

Procedimiento

1. Abra el archivo `mqwebuser.xml`. Este archivo se encuentra en una de las ubicaciones siguientes:
   - ULW

     en UNIX, Linux, and Windows: *MQ_DATA_PATH*/web/installations/*installationName*/servers/mqweb
   - z/OS

     en z/OS: *WLP_user_directory*/servers/mqweb

     donde *directorio_usuario_WLP* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición del servidor mqweb.
2. Añada o descomente el código siguiente en el archivo `mqwebuser.xml`:

```
<virtualHost allowFromEndpointRef="defaultHttpEndpoint" id="default_host">
        <hostAlias>localhost:9080</hostAlias>
</virtualHost>
```

3. Edite el campo **<hostAlias>** , insertando la combinación de nombre de host y puerto que desea permitir.

   Esta combinación puede ser el nombre de host y el nombre de puerto que ha utilizado en la configuración del servidor mqweb. Por ejemplo, si utiliza la configuración predeterminada de `localhost:9443`, es posible que desee utilizar `localhost:9443` en el campo **<hostAlias>** .

   Si es necesario, puede añadir varios campos **<hostAlias>** en las etiquetas **<virtualHost>** para permitir más combinaciones de nombre de host y puerto. Por ejemplo, para permitir las cabeceras de host que utilizan un puerto HTTP, así como las cabeceras de host que utilizan el puerto HTTPS.

V 9.1.0 Auditoría

Los registros de auditoría de las operaciones realizadas en IBM MQ Console y REST API se pueden producir habilitando los sucesos de mandato y de configuración del gestor de colas, y en UNIX, Linux, and Windows los cambios de estado significativos se registran en los archivos de registro del servidor mqweb.

Cambios de estado significativos

ULW

En UNIX, Linux, and Windows, IBM MQ Console registra los cambios de estado significativos como mensajes en los registros del servidor mqweb. Cada mensaje indica el nombre de principal autenticado que ha solicitado la operación.

Los cambios de estado significativos, por ejemplo cuando se crean, inician, finalizan o suprimen gestores de colas, se registran en los archivos `messages.log` y `console.log` del servidor mqweb a nivel de registro [AUDIT]. Cada entrada de registro indica el nombre de principal autenticado que ha solicitado la operación.

Los archivos `messages.log` y `console.log` se pueden encontrar en la ubicación siguiente:

- ULW en UNIX, Linux, and Windows:
  MQ_DATA_PATH\web\installations*installationName*\servers\mqweb\logs

Para obtener más información sobre cómo configurar los niveles de registro del servidor mqweb, consulte Configuración del registro.

Sucesos de mandato y de configuración

Opcionalmente, puede habilitar los sucesos de mandato y de configuración en el gestor de colas para proporcionar información sobre la mayor parte de la actividad de IBM MQ Console y REST API. Por ejemplo, la creación de canales y la consulta de colas generan sucesos de mandato y de configuración. Para obtener más información sobre cómo habilitar sucesos de mandato y de configuración, consulte Control de sucesos de configuración, mandato y registrador.

Para estos mensajes de sucesos de mandato y configuración, el campo MQIACF_EVENT_ORIGIN se establece en MQEVO_REST y el campo MQCACF_EVENT_APPL_IDENTITY indica los 32 primeros caracteres del nombre de principal autenticado. Si un usuario tiene el rol **MQWebAdminRO** o **MQWebAdmin**, el campo MQCACF_EVENT_USER_ID informa del ID de usuario del servidor mqweb, no el nombre de usuario del principal que ha emitido el mandato. Sin embargo, si el usuario tiene el rol **MQWebUser**, MQCACF_EVENT_USER_ID informa del nombre de usuario del principal que ha emitido el mandato.

Conceptos relacionados

"Auditoría" en la página 465
Puede comprobar las intrusiones de seguridad, o intentos de intrusión, mediante mensajes de sucesos. También puede comprobar la seguridad del sistema utilizando IBM MQ Explorer.

Consideraciones de seguridad para el iniciador de canal de IBM MQ Console y REST API en z/OS

IBM MQ Console y REST API tienen características de seguridad que controlar si un usuario puede emitir, visualizar o modificar mandatos. Los mandatos se pasan al gestor de colas y, a continuación, se utiliza la seguridad del gestor de colas para controlar si el usuario está autorizado para emitir el mandato para dicho gestor de colas específico.

Procedimiento

1. Asegúrese de que el ID de usuario de la tarea iniciada del servidor mqweb tiene las autorizaciones apropiadas para emitir determinados mandatos PCF y acceder a determinadas colas. Para obtener más información, consulte "Autorización necesaria para el ID de usuario de tarea iniciada del servidor mqweb" en la página 522.
2. Asegúrese de que todos los usuarios a los que se les ha otorgado el rol `MQWebUser` tienen las autorizaciones apropiadas.

   Los usuarios de IBM MQ Console y REST API asignados al rol `MQWebUser` funcionan bajo el contexto de seguridad del principal. Estos ID de usuario solo pueden realizar operaciones para las que se les ha otorgado autorización para realizar en el gestor de colas y deben tener acceso a las mismas colas de sistema que el espacio de direcciones de servidor mqweb.

   Al ID de usuario de la tarea iniciada del servidor mqweb se les debe haber otorgado acceso de usuario alternativo a todos los usuarios asignados al rol `MQWebUser`.

   Si desea más información sobre cómo otorgar las autorizaciones apropiadas para los usuarios con el rol `MQWebUser`, consulte "Acceso a recursos de IBM MQ necesarios para utilizar la MQ Console o la REST API" en la página 522.
3. Opcional: Configure TLS para la IBM MQ Console y la REST API. Para obtener más información, consulte "Configuración de TLS para REST API y IBM MQ Console en z/OS" en la página 523.

z/OS Autorización necesaria para el ID de usuario de tarea iniciada del servidor mqweb

En z/OS, el ID de usuario de la tarea iniciada del servidor mqweb requiere determinadas autorizaciones para emitir mandatos PCF y acceder a recursos del sistema.

El ID de usuario de la tarea iniciada del servidor mqweb necesita:

- Un identificador de usuario (UID) de z/OS UNIX para poder utilizar z/OS UNIX System Services.
- Acceder a los conjuntos de datos `hlq.SCSQAUTH` y `hlq.SCSQANL*` en la instalación de IBM MQ.
- Acceso de lectura a los archivos de instalación de IBM MQ en z/OS UNIX System Services.
- Acceso de lectura y escritura al directorio de usuarios de Liberty creado por el script **crtmqweb**.
- Autorización para conectarse al gestor de colas. Otorgue al ID de usuario de la tarea iniciada del servidor acceso *READ* al perfil `hlq.BATCH` en la clase MQCONN.
- Autorización para emitir mandatos IBM MQ y acceder a determinadas colas. Estos detalles se describen en "IBM MQ Console - perfiles de seguridad de mandato necesarios" en la página 230, "Seguridad de colas del sistema" en la página 207 y "Perfiles para la seguridad de contexto" en la página 218.
- Autorización para suscribirse al tema SYSTEM.FTE , para utilizar REST API para MFT. Otorgue al ID de usuario de la tarea iniciada del servidor mqweb acceso *ALTER* al perfil `hlq.SUBSCRIBE.SYSTEM.FTE` en la clase MXTOPIC.
- Si está configurando un registro SAF, acceda a distintos perfiles de seguridad. Consulte "Configuración de un registro SAF para la IBM MQ Console y REST API" en la página 506 para obtener más información.

Autenticación de conexión

Si el gestor de colas se ha configurado para requerir todas las aplicaciones por lotes, proporcione un ID de usuario y una contraseña válidos, estableciendo CHKLOCL(REQUIRED), debe proporcionar al ID de usuario de la tarea iniciada del servidor mqweb *UPDATE* acceso al perfil `hlq.BATCH` en la clase MQCONN.

Esta autorización hace que la autenticación de conexión funcione en la modalidad CHKLOCL(OPTIONAL) para el ID de usuario de la tarea iniciada del servidor mqweb.

Si no ha configurado el gestor de colas para que requiera todas las aplicaciones por lotes proporcione un ID de usuario y una contraseña válidos, es suficiente proporcionar al ID de usuario que inicia la tarea del servidor mqweb acceso *READ* al perfil `hlq.BATCH` en la clase MQCONN.

Para obtener más información sobre CHCKLOCL, consulte "Utilización de CHCKLOCL en aplicaciones enlazadas localmente" en la página 197.

Acceso a recursos de IBM MQ necesarios para utilizar la MQ Console o la REST API

Las operaciones realizadas en MQ Console o REST API por un usuario con el rol `MQWebUser` tienen lugar en el contexto de seguridad del usuario.

Acerca de esta tarea

Consulte "Roles en IBM MQ Console y REST API" en la página 509 para obtener más información sobre los roles en MQ Console y REST API.

Utilice el procedimiento siguiente para otorgar a un usuario con el rol `MQWebUser` acceso a los recursos del gestor de colas necesarios para utilizar MQ Console o REST API.

Procedimiento

1. Otorgue al ID de usuario `mqweb server started task` acceso de usuario alternativo a cada ID de usuario en el rol `MQWebUser`.

Realice esta tarea en cada uno de los gestores de colas que administrarán los usuarios mediante MQ Console o REST API.

Puede utilizar los siguientes mandatos RACF de ejemplo para otorgar al ID de usuario de `mqweb server started task` acceso de usuario alternativo a un usuario con el rol `MQWebUser` :

```
RDEFINE MQADMIN hlq.ALTERNATE.USER.userId UACC(NONE)
PERMIT hlq.ALTERNATE.USER.userId CLASS(MQADMIN) ACCESS(UPDATE) ID(mqwebUserId)
SETROPTS RACLIST(MQADMIN) REFRESH
```

donde:

hlq
Es el prefijo de perfil, que puede ser el nombre del gestor de colas o el nombre del grupo de compartición de colas

userId
Es el usuario con el rol `MQWebUser`

mqwebUserId
Es el ID de usuario de `mqweb server started task`

Nota: Si utiliza seguridad que combina mayúsculas y minúsculas, utilice la clase MXADMIN y no la clase MQADMIN.

2. Otorgue a cada usuario con el rol `MQWebUser` acceso a las colas del sistemas que son necesarios para utilizar la MQ Console y la REST API.

   Para ello, tanto para SYSTEM.ADMIN.COMMAND.QUEUE como para SYSTEM.REST.REPLY.QUEUE, proporcione a cada usuario acceso UPDATE a las clases MQQUEUE o MXQUEUE, según si se utiliza o no seguridad que combina mayúsculas y minúsculas.

   Debe hacer esto en cada gestor de colas que el usuario administrará a través de la REST API, incluidos los gestores de colas remotos administrados a través de la pasarela deadministrative REST API.
3. Para permitir que un usuario con el rol `MQWebUser` administre los gestores de colas remotos, otorgue al usuario acceso UPDATE al perfil en la clase MQQUEUE o MXQUEUE, protegiendo la cola de transmisión utilizada para enviar mandatos a un gestor de colas remoto. Tenga en cuenta que debe proporcionar al usuario el acceso UPDATE en el gestor de colas de pasarela.

   En el gestor de colas remoto, otorgue acceso al mismo usuario para la colocación en la cola de transmisión utilizada para devolver mensajes de respuesta al gestor de colas de la pasarela.
4. Otorgue a los usuarios del rol `MQWebUser` acceso a cualquier otro recurso necesario para realizar las operaciones soportadas por la MQ Console y la REST API.

   El acceso necesario para:

   - Realizar operaciones en la REST API, se describe en las secciones *Requisitos de seguridad* de los recursos de la REST API individuales.
   - Emitir mandatos mediante MQ Console se describe en “IBM MQ Console - perfiles de seguridad de mandato necesarios” en la página 230

V 9.1.0 Configuración de TLS para REST API y IBM MQ Console en z/OS

En z/OS, puede configurar el servidor mqweb para utilizar un conjunto de claves de RACF para almacenar certificados para conexiones seguras con TLS, y la autenticación de certificado de cliente.

Antes de empezar

Debe ser un usuario que tenga acceso de escritura en el archivo `mqwebuser.xml` y la autoridad para trabajar con conjuntos de claves SAF, para completar este procedimiento.

Acerca de esta tarea

La configuración de servidor mqweb predeterminada utiliza almacenes de claves Java para el servidor y certificados de confianza. En z/OS, puede configurar el servidor mqweb para que utilice un conjunto de claves de RACF, en lugar de los almacenes de claves de Java. El servidor también se puede configurar para permitir a los usuarios autenticarse utilizando un certificado de cliente.

Consulte Liberty: Conjuntos de claves si desea información sobre cómo utilizar conjuntos de claves de RACF en Liberty.

Siga este procedimiento para configurar el servidor mqweb para que utilice un conjunto de claves de RACF y, si lo desea, configurar la autenticación de certificado de cliente.

Procedimiento

1. Cree un certificado de entidad emisora de certificados (CA), que se utilizará para firmar el certificado de servidor. Por ejemplo, especifique el mandato RACF siguiente:

```
RACDCERT GENCERT
  CERTAUTH
  SUBJECTSDN(CN('mqweb Certification Authority')
    O('IBM')
    OU('MQ'))
  SIZE(2048)
  WITHLABEL('mqwebCertauth')
```

2. Cree un certificado de servidor, firmado con el certificado de CA creado en el paso 1, especificando el mandato siguiente:

```
RACDCERT ID(mqwebUserId) GENCERT
  SUBJECTSDN(CN('hostname')
   O('IBM')
   OU('MQ'))
   SIZE(2048)
   SIGNWITH (CERTAUTH LABEL('mqwebCertauth'))
  WITHLABEL('mqwebServerCert')
```

   donde *mqwebUserId* es el ID de usuario de la tarea iniciada del servidor mqweb, y *hostname* es el nombre de host del servidor mqweb.

3. Conecte el certificado de CA y el certificado de servidor a un conjunto de claves SAF especificando los mandatos siguientes:

```
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebCertauth') CERTAUTH)
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebServerCert'))
```

   donde *mqwebUserId* es el ID de usuario de la tarea iniciada del servidor mqweb, y *keyring* es el nombre del conjunto de claves que desea utilizar.

4. Exporte el certificado de CA a un archivo CER especificando el mandato siguiente:

```
RACDCERT CERTAUTH EXPORT(LABEL('mqwebCertauth'))
        DSN('hlq.CERT.MQWEBCA')
        FORMAT(CERTDER)
        PASSWORD('password')
```

5. Envíe por FTP el certificado de CA exportado en binario a la estación de trabajo e impórtelo en el navegador como un certificado de autoridad emisora de certificados.
6. Opcional: Si desea configurar la autenticación de certificado de cliente, cree y exporte un certificado de cliente.
   a) Cree un certificado de entidad emisora de certificados (CA), que se utilizará para firmar el certificado de cliente. Por ejemplo, especifique el mandato RACF siguiente:

```
RACDCERT GENCERT
```

```
  CERTAUTH
  SUBJECTSDN(CN('mqweb User CA')
   O('IBM')
   OU('MQ'))
   SIZE(2048)
  WITHLABEL('mqwebUserCertauth')
```

b) Conecte el certificado de CA a un conjunto de claves SAF especificando el mandato siguiente:

```
RACDCERT ID(mqwebUserId) CONNECT(RING(keyring) LABEL('mqwebUserCertauth') CERTAUTH)
```

donde *mqwebUserId* es el ID de usuario de la tarea iniciada del servidor mqweb, y *keyring* es el nombre del conjunto de claves que desea utilizar.

c) Cree un certificado de cliente, firmado con el certificado de CA. Por ejemplo, especifique el mandato siguiente:

```
RACDCERT ID(clientUserId) GENCERT
  SUBJECTSDN(CN('clientUserId')
     O('IBM')
     OU('MQ'))
   SIZE(2048)
   SIGNWITH (CERTAUTH LABEL('mqwebUserCertauth'))
  WITHLABEL('userCertLabel')
```

donde *clientUserId* es el nombre de usuario.

El método utilizado para correlacionar un certificado con un principal depende del tipo de registro de usuarios configurado:

- Si está utilizando un registro básico, el campo Nombre común en el certificado se compara con el usuario del registro.
- Si está utilizando un registro SAF, y el certificado está en la base de datos RACF, se utiliza el propietario del certificado, especificado con el parámetro **ID** cuando se crea el certificado.
- Si está utilizando un registro LDAP, el nombre distinguido completo del certificado se compara con el registro LDAP.

d) Exporte el certificado de cliente a un archivo PKCS #12 especificando el mandato siguiente:

```
RACDCERT ID(mqwebUserId) EXPORT(LABEL('userCertLabel')) PASSWORD('password')
DSN('hlq.USER.CERT')
```

e) Envíe por FTP el certificado exportado en binario a la estación de trabajo. Para utilizar el certificado de cliente con la IBM MQ Console, impórtelo en el navegador web utilizado para acceder a la IBM MQ Console como un certificado personal.

7. Edite el archivo *WLP_user_directory*/servers/mqweb/mqwebuser.xml, donde *WLP_user_directory* es el directorio que se ha especificado cuando se ejecutó el script **crtmqweb** para crear la definición de servidor mqweb.

Realice los cambios siguientes para configurar el servidor mqweb para que utilice un conjunto de claves de RACF:

a) Elimine o comente la línea siguiente:

```
<sslDefault sslRef="mqDefaultSSLConfig"/>
```

b) Añada las sentencias siguientes:

```
<keyStore id="defaultKeyStore" filebased="false" location="safkeyring://mqwebUserId/
keyring"
          password="password" readOnly="true" type="JCERACFKS" />
<ssl id="thisSSLConfig" keyStoreRef="defaultKeyStore" sslProtocol="TLSv1.2"
          serverKeyAlias="mqwebServerCert" clientAuthenticationSupported="true" />
<sslDefault sslRef="thisSSLConfig"/>
```

donde:

- *mqwebUserId* es el ID de usuario de la tarea iniciada del servidor mqweb.
- *keyring* es el nombre del conjunto de claves de RACF.
- *mqwebServerCert* es la etiqueta del certificado de servidor mqweb.

Notas: El valor de **keyStore password** se ignora.

8. Reinicie el servidor mqweb deteniendo y reiniciando la tarea iniciada del servidor mqweb.
9. Opcional: Utilice el certificado de cliente para la autenticación:
   - Para utilizar el certificado de cliente con la IBM MQ Console, especifique el URL para la MQ Console en el navegador web donde ha instalado el certificado de cliente.
   - Para utilizar el certificado de cliente con la API REST, proporcione el certificado de cliente con cada solicitud REST.

   Notas:

   a. Si solo está utilizando certificados para la autenticación en IBM MQ Console, el navegador mostrará una lista de certificados que puede seleccionar.

   b. Si desea utilizar un certificado diferente, es posible que tenga que cerrar y reiniciar el navegador.

   c. Si está utilizando certificados de cliente que no están en la base de datos RACF, puede utilizar el filtrado de nombres de certificado de RACF para correlacionar atributos de certificados con un ID de usuario. Por ejemplo:

   ```
   RACDCERT ID(DEPT3USR) MAP SDNFILTER(OU=DEPT1.C=US)
   ```

   correlaciona los certificados con un nombre distinguido de asunto que contiene OU=DEPT1 y C=US con el ID de usuario DEPT3USR.

Resultados

Ha configurado una interfaz TLS para IBM MQ Console y REST API.

ULW Gestión de claves y certificados en UNIX, Linux, and Windows

Utilice el mandato `runmqckm` (UNIX y Windows), y el mandato `runmqakm` (UNIX, Linux, and Windows) para gestionar claves, certificados y solicitudes de certificado.

El mandato runmqckm

El mandato **runmqckm** está disponible en UNIX y Windows.

El mandato **runmqckm** proporciona funciones que son similares a las de iKeyman, que se describen en "Protección de IBM MQ" en la página 5.

Para utilizar el mandato **runmqckm** , asegúrese de que las variables de entorno del sistema se hayan configurado correctamente ejecutando el mandato **setmqenv** .

V 9.1.0 El mandato **runmqckm** requiere que el componente IBM MQ JRE esté instalado. Si el componente no está instalado, puede utilizar el mandato **runmqackm** en su lugar.

El mandato runmqakm

El mandato **runmqakm** está disponible en UNIX, Linux y Windows.

Para utilizar el mandato **runmqaqm** , asegúrese de que las variables de entorno del sistema se hayan configurado correctamente ejecutando el mandato **setmqenv** .

Si necesita gestionar certificados TLS de forma que cumplan los estándares FIPS, utilice el mandato **runmqakm** en lugar de los mandatos **runmqckm**. Esto se debe a que el mandato **runmqakm** soporta un cifrado más potente.

Utilice los mandatos `runmqckm` y `runmqakm` para realizar lo siguiente:

- Crear el tipo de archivos de base de datos de claves CMS que necesita IBM MQ
- Crear solicitudes de certificado
- Importar certificados personales
- Importar certificados CA
- Gestionar certificados autofirmados

Información relacionada
Keytool

ULW Mandatos runmqckm y runmqakm en UNIX, Linux, and Windows

En esta sección se describen los mandatos runmqckm y runmqakm según el objeto del mandato.

Las diferencias principales entre los dos mandatos son las siguientes:

- ULW **runmqakm**
  - Está disponible en UNIX, Linux y Windows.
  - Admite la creación de certificados y solicitudes de certificado con claves públicas Elliptic Curve, mientras que el mandato **runmqckm** no.
  - Admite un cifrado más potente del archivo del repositorio de claves que el mandato **runmqckm** mediante el parámetro **-strong**.
  - Se ha certificado como compatible con FIPS 140-2 y se puede configurar para que funcione de forma compatible con FIPS, utilizando el parámetro **-fips**, a diferencia del mandato **runmqckm**.
- Windows UNIX **runmqckm**
  - Está disponible en UNIX y Windows.
  - Da soporte a los formatos de archivo de repositorio de claves JKS y JCEKS, mientras que el mandato **runmqakm** no.

Atención: V 9.1.0 El mandato **runmqckm** requiere la instalación de la característica IBM MQ Java runtime environment (JRE).

Cada mandato especifica al menos un *objeto*. Los mandatos para las operaciones de dispositivos PKCS #11 pueden especificar objetos adicionales. Los mandatos para los objetos de base de datos de claves, certificado y solicitud de certificado especifican además una *acción*. El objeto puede ser uno de los siguientes:

-keydb
Las acciones se aplican a una base de datos de claves

-cert
Las acciones se aplican a un certificado

-certreq
Las acciones se aplican a una solicitud de certificado

-help
Muestra la ayuda

-versión
Muestra información de la versión

Los subtemas siguientes describen las acciones que se pueden realizar en cualquier base de datos de claves, certificado y objetos de solicitud de certificado; consulte “opciones runmqckm y runmqakm en UNIX, Linux, and Windows” en la página 537 para ver una descripción de las opciones de estos mandatos.

ULW Mandatos para una base de datos de claves CMS sólo en UNIX, Linux, and Windows

Puede utilizar los mandatos **runmqckm** y **runmqakm** para gestionar claves y certificados para una base de datos de claves CMS.

-keydb -changepw
Cambiar la contraseña de una base de datos de claves CMS:

```
-keydb -changepw -db filename -pw password -new_pw new_password


-stash
```

-keydb -create
Crear una base de datos de claves CMS:

```
-keydb -create -db filename
-pw password -type cms -expire days -stash
```

-keydb -stashpw
Ocultar la contraseña de una base de datos de claves CMS en un archivo:

```
-keydb -stashpw -db filename
-pw password
```

-cert -getdefault

Nota: El certificado predeterminado no está soportado por IBM MQ 8.0. Debería utilizar la configuración de la etiqueta de certificado, tal como se describe en "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26.

Obtener el certificado personal predeterminado:

```
-cert -getdefault -db filename
-pw password
```

-cert -modify
Modificar un certificado.

Nota: Actualmente, el único campo que se puede modificar es el campo Certificado de confianza.

```
-cert -modify -db filename
-pw password -label label
-trust enable|disable
```

-cert -setdefault

Nota: El certificado predeterminado no está soportado por IBM MQ 8.0 o posterior. Debería utilizar la configuración de la etiqueta de certificado, tal como se describe en "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26.

Establecer el certificado personal predeterminado:

```
-cert -setdefault -db filename
-pw password -label label
```

ULW Mandato para bases de datos de claves CMS o PKCS #12 en UNIX, Linux, and Windows

Puede utilizar los mandatos runmqckm y runmqakm para gestionar las claves y los certificados para una base de datos de claves CMS o una base de datos de claves PKCS #12.

Nota: IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.

Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.

-keydb -changepw
Cambiar la contraseña de una base de datos de claves:

```
-keydb -changepw -db filename -pw password -new_pw
new_password -expire days
```

-keydb -convert
convierta la base de datos de claves de un formato a otro:

```
-keydb -convert -db filename -pw password
-old_format cms | pkcs12 -new_format cms
```

-keydb -create
Crear una base de datos de claves:

```
-keydb -create -db filename -pw password -type cms
| pkcs12
```

-keydb -delete
Suprimir una base de datos de claves:

```
-keydb -delete -db filename -pw password
```

-keydb -list
Listar los tipos de base de datos de claves soportados actualmente:

```
-keydb -list
```

-cert -add
Añadir un certificado de un archivo a una base de datos de claves:

```
-cert -add -db filename -pw password -label label
-file filename
-format ascii | binary
```

-cert -create
Crear un certificado autofirmado:

```
-cert -create -db filename -pw password -label label
-dn distinguished_name
-size 1024 | 512 -x509version 3 | 1
| 2
-expire days -sig_alg MD2_WITH_RSA | MD2WithRSA
|
MD5_WITH_RSA | MD5WithRSA
|
SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA
```

```
|
SHA2WithRSA | SHA384_WITH_RSA
|
SHA384WithRSA | SHA512_WITH_RSA
|
SHA512WithRSA | SHA_WITH_DSA
|
SHA_WITH_RSA | SHAWithDSA
|
SHAWithRSA
```

-cert -delete
Suprimir un certificado:

```
-cert -delete -db filename -pw password -label label
```

-cert -details
Listar la información detallada de un certificado específico:

```
-cert -details -db filename -pw password -label label
```

-cert -export
Exportar un certificado personal y su clave privada asociada de una base de datos de claves a un archivo PKCS#12, o a otra base de datos de claves:

```
-cert -export -db filename -pw password -label label
-type cms | pkcs12
-target filename -target_pw password -target_type
cms | pkcs12
```

-cert -extract
Extraer un certificado de una base de datos de claves:

```
-cert -extract -db filename -pw password -label label
-target filename
-format ascii | binary
```

-cert -import
Importar un certificado personal de una base de datos de claves:

```
-cert -import -file filename -pw password -type
pkcs12 -target filename
-target_pw password -target_type cms -label
label
```

La opción `-label` es necesaria y especifica la etiqueta del certificado que se va a importar de la base de datos de claves origen.

La opción `-new_label` es opcional y permite asignar al certificado importado una etiqueta diferente en la base de datos de claves de destino de la etiqueta en la base de datos origen.

-cert -list
Listar todos los certificados en una base de datos de claves:

```
-cert -list all | personal | CA
-db filename -pw password
```

-cert -receive
Recibir un certificado de un archivo:

```
-cert -receive -file filename -db filename -pw password
```

```
-format ascii | binary -default_cert yes |
no
```

-cert -sign

Firmar un certificado:

```
-cert -sign -db filename -file filename -pw password
-label label -target filename
-format ascii | binary -expire days
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

-certreq -create

Crear una solicitud de certificado:

```
-certreq -create -db filename -pw password
-label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA |
MD5_WITH_RSA | MD5WithRSA |
SHA1WithDSA | SHA1WithRSA |
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

-certreq -delete

Suprimir una solicitud de certificado:

```
-certreq -delete -db filename -pw password -label
label
```

-certreq -details

Listar la información detallada de una solicitud de certificado específica:

```
-certreq -details -db filename -pw password -label
label
```

Listar la información detallada para una solicitud de certificado y mostrar la solicitud de certificado completa:

```
-certreq -details -showOID -db filename
-pw password -label label
```

-certreq -extract

Extraer una solicitud de certificado de una base de datos de solicitudes de certificado en un archivo:

```
-certreq -extract -db filename -pw password
-label label -target filename
```

-certreq -list

Listar todas las solicitudes de certificado de la base de datos de solicitudes de certificado:

```
-certreq -list -db filename -pw password
```

-certreq -recreate
Vuelva a crear una solicitud de certificado:

```
-certreq -recreate -db filename -pw password
-label label -target filename
```

ULW Mandatos para operaciones de dispositivo criptográfico en UNIX, Linux, and Windows

Puede utilizar los mandatos runmqckm y runmqakm para gestionar las claves y los certificados para las operaciones de dispositivo criptográfico.

Nota: IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.

Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.

-keydb -changepw
Cambiar la contraseña de un dispositivo de cifrado:

```
-keydb -changepw -crypto module_name -tokenlabel token_label
-pw password -new_pw new_password
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-keydb -list
Listar los tipos de base de datos de claves soportados actualmente:

```
-keydb -list
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -add
Añadir un certificado de un archivo a un dispositivo de cifrado:

```
-cert -add -crypto module_name -tokenlabel token_label
-pw password -label label -file filename -format
ascii | binary
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -create
Crear un certificado autofirmado en un dispositivo de cifrado:

```
-cert -create -crypto module_name -tokenlabel token_label

-pw password -label label -dn distinguished_name
-size 1024 | 512
-x509version 3 | 1 | 2 -default_cert no
| yes -expire days
-sig_alg MD2_WITH_RSA | MD2WithRSA |
MD5_WITH_RSA | MD5WithRSA |
SHA1WithDSA | SHA1WithRSA |
SHA256_WITH_RSA | SHA256WithRSA |
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

Nota: No puede importar un certificado que contenga varios atributos OU (unidades organizativa) en el nombre distinguido.

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -delete

Suprimir un certificado en un dispositivo de cifrado:

```
-cert -delete -crypto module_name -tokenlabel token_label
-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -details

Listar la información detallada para un certificado específico en un dispositivo de cifrado:

```
-cert -details -crypto module_name -tokenlabel token_label

-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

Listar la información detallada y mostrar el certificado completo para un certificado específico en un dispositivo de cifrado:

```
-cert -details -showOID -crypto module_name -tokenlabel
token_label
-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca

PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -extract
Extraer un certificado de una base de datos de claves:

```
-cert -extract -crypto module_name -tokenlabel token_label
-pw password -label label -target filename
-format ascii | binary
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -import
Importar un certificado a un dispositivo de cifrado con soporte de base de datos de claves secundaria:

```
-cert -import -db filename -pw password -label label
-type cms
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

```
-cert -import -db filename -pw password -label label
-type cms
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password -fips
```

Importar un certificado PKCS #12 a un dispositivo de cifrado con soporte de base de datos de claves secundaria:

```
-cert -import -file filename -pw password -type pkcs12
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

```
-cert -import -file filename -pw password -type pkcs12
-crypto module_name -tokenlabel token_label -pw
password
-secondaryDB filename -secondaryDBpw password -fips
```

Nota: No puede importar un certificado que contenga varios atributos OU (unidades organizativa) en el nombre distinguido.

-cert -list
Listar todos los certificados de un dispositivo de cifrado:

```
-cert -list all | personal | CA
-crypto module_name -tokenlabel token_label -pw
password
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-cert -receive
Recibir un certificado de un archivo en un dispositivo de cifrado con soporte de base de datos de claves secundaria:

```
-cert -receive -file filename -crypto module_name -tokenlabel
token_label
-pw password -default_cert yes | no
-secondaryDB filename -secondaryDBpw password -format
ascii | binary
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

Utilizando el mandato **runmqakm**:

-certreq -create
Crear una solicitud de certificado en un dispositivo de cifrado:

```
-certreq -create -crypto module_name -tokenlabel token_label

-pw password -label label -dn distinguished_name
-size 1024 | 512 -file filename
-sig_alg MD2_WITH_RSA | MD2WithRSA | MD5_WITH_RSA
|
MD5WithRSA | SHA1WithDSA | SHA1WithRSA
|
SHA256_WITH_RSA | SHA256WithRSA
SHA2WithRSA | SHA384_WITH_RSA |
SHA384WithRSA | SHA512_WITH_RSA |
SHA512WithRSA | SHA_WITH_DSA |
SHA_WITH_RSA | SHAWithDSA |
SHAWithRSA
```

Nota: No puede importar un certificado que contenga varios atributos OU (unidades organizativa) en el nombre distinguido.

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-certreq -delete
Suprimir una solicitud de certificado de un dispositivo de cifrado:

```
-certreq -delete -crypto module_name -tokenlabel token_label
-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-certreq -details
Listar la información detallada de una solicitud de certificado específica en un dispositivo de cifrado:

```
-certreq -details -crypto module_name -tokenlabel token_label
-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

Listar la información detallada sobre una solicitud de certificado y mostrar la solicitud de certificado completa en un dispositivo de cifrado:

```
-certreq -details -showOID -crypto module_name -tokenlabel
token_label
-pw password -label label
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-certreq -extract
Extraer una solicitud de certificado de una base de datos de solicitudes de certificado de un dispositivo de cifrado en un archivo:

```
-certreq -extract -crypto module_name -tokenlabel token_label
-pw password -label label -target filename
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

-certreq -list
Listar todas las solicitudes de certificado de la base de datos de solicitudes de certificado en un dispositivo de cifrado:

```
-certreq -list -crypto module_name -tokenlabel token_label
-pw password
```

Si está utilizando claves o certificados almacenados en hardware de cifrado PKCS #11, tenga en cuenta que **runmqckm** y **strmqikm** son programas de 64 bits. Los módulos externos necesarios para el soporte de PKCS #11 se cargarán en un proceso de 64 bits, por lo que debe tener una biblioteca PKCS #11 de 64 bits instalada para la administración de hardware de cifrado. Las plataformas Windows y Linux x86 de 32 bits son las únicas excepciones, puesto que los programas **strmqikm** y **runmqckm** son de 32 bits en esas plataformas.

ULW opciones runmqckm y runmqakm en UNIX, Linux, and Windows

Puede utilizar las opciones de línea de mandatos **runmqckm** (iKeycmd) y **runmqakm** para gestionar claves, certificados y solicitudes de certificado.

ULW El mandato runmqakm está disponible en UNIX, Linux, and Windows.

Windows UNIX El mandato runmqckm está disponible en UNIX y Windows.

Nota: IBM MQ no da soporte a los algoritmos SHA-3 o SHA-5. Puede utilizar los nombres del algoritmo de firma digital SHA384WithRSA y SHA512WithRSA porque ambos algoritmos son miembros de la familia SHA-2.

Los nombres de algoritmo de firma digital SHA3WithRSA y SHA5WithRSA están en desuso porque son una forma abreviada de SHA384WithRSA y SHA512WithRSA respectivamente.

El significado de una opción puede depender del objeto y de la acción especificados en el mandato.

*Tabla 90. Opciones que se pueden utilizar con **runmqckm** y **runmqakm***

Parámetro	Descripción
-create	Opción para crear una base de datos de claves.
-crypto	Nombre del módulo para gestionar un dispositivo de cifrado PKCS #11. El valor después de **-crypto** es opcional si especifica el nombre de módulo en el archivo de propiedades. Si está utilizando certificados o claves almacenados en hardware criptográfico PKCS #11 , tenga en cuenta que **runmqckm** y **strmqikm** se ejecutan utilizando la máquina virtual Java (JVM) proporcionada con la instalación de IBM MQ . Los módulos externos necesarios para el soporte PKCS #11 se cargarán en el proceso JVM, por lo tanto, debe tener instalada una biblioteca PKCS #11 para la administración del hardware de cifrado que coincida con el bit de la JVM, y debe especificar esta biblioteca en **runmqckm** o **strmqikm**.
-db	Nombre de vía de acceso totalmente calificada de una base de datos de claves.
-default_cert	Establece un certificado como el certificado predeterminado. El valor puede ser Yes o No. El valor predeterminado es no.
-dn	Nombre distinguido X.500. El valor es una serie entre comillas dobles, por ejemplo "CN=John Smith,O=IBM,OU=Test,C=GB". Tenga en cuenta que solo los atributos O y C son necesarios. La especificación de un nombre común (CN) es opcional.
-encryption	Potencia del cifrado utilizado en el mandato de exportación de certificado. El valor puede ser STRONG o WEAK. El valor predeterminado es STRONG.

*Tabla 90. Opciones que se pueden utilizar con **runmqckm** y **runmqakm** (continuación)*	
Parámetro	**Descripción**
-expire	Tiempo de caducidad en días de un certificado o una contraseña de la base de datos. El valor predeterminado es 365 días para una contraseña de certificado. No hay ningún tiempo predeterminado para una contraseña de base de datos: utilice el parámetro **-expire** para establecer explícitamente un tiempo de caducidad de la contraseña de la base de datos.
-file	Nombre de archivo de un certificado o una solicitud de certificado.
-fips	Especifica que el mandato se ejecuta en modalidad FIPS. Cuando se utiliza la modalidad FIPS, el componente ICC utiliza algoritmos que se han validado mediante FIPS 140-2. Si el componente ICC no se inicializa en modalidad FIPS, el mandato **runmqakm** no se ejecuta correctamente.
-format	Formato de un certificado. El valor puede ser `ascii` para ASCII Base64_encoded o `binary` para datos DER binarios. El valor predeterminado es `ascii`.
-label	Etiqueta adjunta a un certificado o una solicitud de certificado. Si el certificado es un certificado personal utilizado para identificar una aplicación cliente o un gestor de colas IBM MQ, la etiqueta debe corresponder al valor de etiqueta de certificado IBM MQ (CERTLABL), si desea más información, consulte "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26.
-new_format	Nuevo formato de la base de datos de claves.
-new_label	Esta opción se utiliza en un mandato de importación de certificado y permite importar un certificado con una etiqueta diferente de la que tenía en la base de datos de claves de origen. Si el certificado es un certificado personal utilizado para identificar una aplicación cliente o un gestor de colas IBM MQ, la etiqueta debe corresponder al valor de etiqueta de certificado IBM MQ (CERTLABL), si desea más información, consulte "Etiquetas de certificados digitales, descripción de los requisitos" en la página 26.
-new_pw	Nueva contraseña de la base de datos.
-old_format	Formato antiguo de la base de datos de claves.
-pw	Contraseña de la base de datos de claves o el archivo PKCS #12.
-secondaryDB	Nombre de una base de datos de claves secundaria para operaciones de dispositivos PKCS #11.
-secondaryDBpw	Contraseña de la base de datos de claves secundaria para operaciones de dispositivos PKCS #11.
-showOID	Muestra el certificado o la solicitud de certificado completos.

*Tabla 90. Opciones que se pueden utilizar con **runmqckm** y **runmqakm** (continuación)*	
Parámetro	**Descripción**
-sig_alg	El algoritmo hash utilizado durante la creación de una solicitud de certificado, un certificado autofirmado, o la firma de un certificado. Este algoritmo hash se utiliza para crear la firma asociada al certificado o solicitud de certificado recién creado. Para **runmqckm**, el valor puede ser, MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA, SHAWithRSA. El valor predeterminado es SHA1WithRSA. Para **runmqakm**, el valor puede 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 o EC_ecdsa_with_SHA512. El valor predeterminado es SHA1WithRSA.
-size	Tamaño de clave. Para **runmqckm**, el valor puede ser 512, 1024, o 2048. El valor predeterminado es 1024 bits. Para **runmqakm**, el valor depende del algoritmo de firma: • Para los algoritmos de firma RSA (el valor predeterminado del algoritmo que se utiliza si no se especifica **-sig_alg**), el valor puede ser 512, 1024, 2048, o 4096. Un tamaño de clave RSA de 512 bits no está permitido si el parámetro **-fips** está habilitado. El tamaño de clave RSA predeterminado es 1024 bits. • Para algoritmos Elliptic Curve, el valor puede ser 256, 384 ó 512. El valor predeterminado del tamaño de clave Elliptic Curve depende del algoritmo de firma. Para SHA256, es 256; para SHA384, es 384; y para SHA512, es 512.
-stash	Almacene la contraseña de la base de datos de claves en un archivo. Solo es aplicable a las bases de datos de tipo CMS y PKCS12. **Nota: -stash** es válido en los mandatos **-keydb** -create para indicar a **runmqckm**/**runmqakm** que cree un archivo de ocultación que contenga la contraseña. Al emitir el mandato `$ runmqakm -help` sólo se listan los parámetros de ayuda de alto nivel.

*Tabla 90. Opciones que se pueden utilizar con **runmqckm** y **runmqakm** (continuación)*	
Parámetro	**Descripción**
-stashed	Indica que la contraseña para la base de datos de claves o el archivo PKCS #12 está en un archivo de ocultación. **Nota:** La opción **-stashed** es válida en llamadas aparte de los mandatos **-keydb** -create. Si no especifica esta opción, debe proporcionar la contraseña utilizando **-pw**. Además, sólo cuando indica al mandato qué tipo de acción está realizando, aparece la ayuda detallada que muestra **-stashed** .
-target	Archivo o base de datos de destino.
-target_pw	Contraseña de la base de datos de claves si **-target** especifica una base de datos de claves.
-target_type	Tipo de base de datos especificado por el operando **-target**. Consulte el parámetro **-type** para ver los valores permitidos.
-tokenLabel	Etiqueta de un dispositivo de cifrado PKCS #11.
-trust	Estado de confianza de un certificado CA. El valor puede ser `enable` o `disable`. El valor predeterminado es `enable`.
-type	Tipo de base de datos. El valor puede ser cualquiera de los valores siguientes: • `cms` para una base de datos de claves CMS • `pkcs12` para un archivo PKCS #12.
-x509version	Versión del certificado X.509 que se va a crear. El valor puede ser 1, 2 o 3. El valor predeterminado es 3.

-rfc3339	Utilice este parámetro para generar la fecha en formato RFC 3339 para el mandato `runmqakm -cert -details`, que tiene el formato siguiente:
	`Not Before : 2015-08-26T08:53:37Z`
	`Not After : 2016-08-26T08:53:37Z`
	Tenga en cuenta que el parámetro **-rfc3339** debe aparecer en el mandato después de los parámetros adicionales:
	`runmqakm -cert -details -db exampleDB -stashed -label certficateLabel -rfc3339`

Nota: Las propiedades proporcionadas con IBM Global Security Kit (GSKit) relacionadas con el parámetro **-seckey** de cifrado de clave simétrica en el programa de utilidad **runmqckm** se ignoran y no están soportadas por IBM MQ.

ULW códigos de error runmqakm en UNIX, Linux, and Windows

Una tabla de los códigos de error numéricos emitidos por runmqakm y lo que significan.

Código de error	Mensaje de error
0	Correcto
1	Se ha producido un error desconocido

Código de error	Mensaje de error
2	Se ha producido un error de codificación/ descodificación ASN.1.
3	Se ha producido un error al inicializar la codificación/descodificación de ASN.1.
4	Se ha producido un error de codificación/ descodificación de ASN.1 debido a un índice fuera de intervalo o a un campo opcional no existente.
5	Se ha producido un error de base de datos.
6	Se ha producido un error al abrir el archivo de base de datos, compruebe la existencia y permisos del archivo.
7	Ha habido un error al volver a abrir el archivo de base de datos.
8	La creación de la base de datos no se ha realizado correctamente.
9	La base de datos ya existe.
10	Ha habido un error al suprimir el archivo de base de datos.
11	No se ha podido abrir la base de datos.
12	Ha habido un error al leer el archivo de base de datos.
13	Ha habido un error al grabar datos en el archivo de base de datos.
14	Se ha producido un error de validación de base de datos.
15	Se ha encontrado una versión no válida de base de datos.
16	Se ha encontrado una contraseña no válida de base de datos.
17	Se ha encontrado un tipo de archivo de base de datos no válido.
18	La base datos especificada está corrupta.
19	La contraseña proporcionada no es válida o la base de datos de claves se ha manipulado o está corrupta.
20	Se ha producido un error en la integridad de entradas de claves de base de datos.
21	Ya existe un certificado duplicado en la base de datos.
22	Ya existe una clave duplicada en la base de datos (ID de registro).
23	Ya existe un certificado con la misma etiqueta en la base de datos de claves.

Código de error	Mensaje de error
24	Ya existe una clave duplicada en la base de datos (Firma).
25	Ya existe una clave duplicada en la base de datos (Certificado no firmado).
26	Ya existe una clave duplicada en la base de datos (Emisor y número de serie).
27	Ya existe una clave duplicada en la base de datos (Información de clave pública de asunto).
28	Ya existe una clave duplicada en la base de datos (CRL no firmada).
29	Se ha utilizado la etiqueta en la base de datos.
30	Ha habido un error de cifrado de contraseña.
31	Ha habido un error relacionado con LDAP. (Este programa no soporta LDAP)
32	Se ha producido un error de cifrado.
33	Se ha producido un error de codificación/ descodificación.
34	Se ha encontrado un algoritmo de cifrado no válido.
35	Se ha producido un error al firmar los datos.
36	Se ha producido un error al verificar los datos.
37	Se ha producido un error al calcular resumen de datos.
38	Se ha encontrado un parámetro de cifrado no válido.
39	Se ha encontrado un algoritmo de cifrado que no está soportado.
40	El tamaño de entrada especificado es mayor que el tamaño de módulo soportado.
41	Se ha encontrado un tamaño de módulo que no se soporta.
42	Se ha producido un error de validación de base de datos.
43	La validación de entrada de claves no se ha ejecutado correctamente.
44	Ya existe un campo de extensión duplicado.
45	No es correcta la versión de la clave.
46	No existe un campo de extensión obligatorio.
47	El periodo de validez no incluye el día actual o no está dentro del periodo de validez del emisor
48	El periodo de validez no incluye el día actual o no está dentro del periodo de validez del emisor.

Código de error	Mensaje de error
49	Se ha producido un error al validar la extensión de uso de claves privadas.
50	No se ha encontrado el emisor de la clave.
51	Falta una extensión de certificado obligatoria.
52	Se ha encontrado una extensión de restricción básica no válida.
53	No se ha realizado correctamente la validación de firma de claves.
54	La clave raíz de la clave no es de confianza.
55	Se ha revocado la clave.
56	Se ha producido un error al validar la extensión de identificador de clave de autoridad.
57	Se ha producido un error al validar la extensión de uso de claves privadas.
58	Se ha producido un error al validar la extensión de nombre alternativo de asunto.
59	Se ha producido un error al validar la extensión de nombre alternativo de emisor.
60	Se ha producido un error al validar la extensión de uso de claves.
61	Se ha encontrado un extensión crítica desconocida.
62	Se ha producido un error al validar las entradas de parejas de claves.
63	Se ha producido un error al validar la CRL.
64	Se ha producido un error mutex.
65	Se ha encontrado un parámetro no válido.
66	Se ha encontrado un error de ubicación de memoria o parámetro nulo.
67	El número o tamaño es demasiado grande o demasiado pequeño.
68	La contraseña antigua no es válida.
69	La contraseña nueva no es válida.
70	La contraseña ha caducado.
71	Se ha producido un error relacionado con hebras.
72	Se ha producido un error al crear hebras.
73	Se ha producido un error cuando una hebra estaba esperando para salir.
74	Se ha producido un error de E/S.
75	Se ha producido un error al cargar la CMS.

Código de error	Mensaje de error
76	Se ha producido un error relacionado con hardware de cifrado.
77	No se ha llamado correctamente a la rutina de inicialización de bibliotecas.
78	La tabla de manejadores de base de datos interna está corrupta.
79	Se ha producido un error de ubicación de memoria.
80	Se ha encontrado una opción que no se reconoce.
81	Se ha producido un error al obtener información de tiempo.
82	Se ha producido un error de creación de mutex.
83	Se ha producido un error al abrir el catálogo de mensajes.
84	Se ha producido un error al abrir el catálogo de mensajes de error.
85	Se ha encontrado un nombre de archivo nulo.
86	Se ha producido un error al abrir archivos, compruebe la existencia y permisos del archivo.
87	Se ha producido un error al abrir archivos para leer.
88	Se ha producido un error al abrir archivos para grabar.
89	No existe dicho fichero.
90	El fichero no puede abrirse por la configuración de permisos.
91	Ha habido un error al grabar datos en los archivos.
92	Se ha producido un error al suprimir archivos.
93	Se han encontrado datos codificados en Base64 no válidos.
94	Se ha encontrado un tipo de mensaje en Base64 no válido.
95	Se ha producido un error al codificar datos con reglas de codificación en Base64.
96	Se ha producido un error al codificar datos en Base64.
97	Se ha producido un error al obtener una etiqueta de nombre distinguido.
98	El campo de nombre común obligatorio está vacío.
99	El campo de nombre región o país obligatorio está vacío.
100	Se ha encontrado un manejador de bases de datos no válido.

Código de error	Mensaje de error
101	No existe la base de datos de claves.
102	No existe la base de datos de parejas de claves solicitada.
103	No existe el archivo de contraseñas.
104	La contraseña nueva es igual que la antigua.
105	No se ha encontrado la clave en la base de datos de claves.
106	No se ha encontrado ninguna clave solicitada.
107	No se ha encontrado CA de confianza.
108	No se ha encontrado ninguna clave para el certificado.
109	No existe clave privada en la base de datos de claves.
110	No existe clave predeterminada en la base de datos de claves.
111	No existe clave privada en el registro de claves.
112	No existe certificado en el registro de claves.
113	No existe ninguna entrada de CRL.
114	Se ha encontrado un nombre de archivo de base de datos de claves no válido.
115	Se ha encontrado un tipo de clave privada que no se reconoce.
116	Se ha encontrado una entrada de nombre distinguido no válido.
117	No se ha encontrado ninguna entrada de clave que tenga la etiqueta de claves especificada.
118	La lista de etiquetas de claves está corrupta.
119	Los datos de entrada no son datos PKCS12 válidos.
120	La contraseña no es válida o los datos PKCS12 están corruptos o se han creado con una versión posterior de PKCS12
121	Se ha encontrado un tipo de exportación de claves que no se reconoce.
122	Se ha encontrado un algoritmo de cifrado basado en contraseñas no soportado.
123	Se ha producido un error al convertir el archivo de conjuntos de claves en una base de datos de claves CMS.
124	Se ha producido un error al convertir la base de datos de conjuntos de CMS en archivo de conjuntos de claves.

Código de error	Mensaje de error
125	Se ha producido un error al crear un certificado para la solicitud de certificados.
126	No se puede crear una cadena completa de emisores.
127	Se han encontrado datos WEBDB no válidos.
128	No hay datos para grabar en el archivo de conjuntos de claves.
129	El número de días introducido supera el periodo de validez permitido.
130	La contraseña es demasiado corta; debe tener al menos {0} caracteres.
131	La contraseña debe contener al menos un dígito numérico.
132	Todos los caracteres de la contraseña son alfabéticos o numéricos.
133	Se ha especificado un algoritmo de firma que no está soportado o no se reconoce.
134	Se ha encontrado un tipo no válido de base de datos.
135	Otro dispositivo PKCS#11 está utilizando la base de datos de claves especificada.
136	No se ha especificado ninguna base de datos de claves secundaria.
137	La etiqueta no existe en el dispositivo PKCS#11.
138	Se necesita contraseña para acceder al dispositivo PKCS#11.
139	No se necesita contraseña para acceder al dispositivo PKCS#11.
140	No se ha podido cargar la biblioteca de cifrado.
141	PKCS#11 no está soportado para esta operación.
142	No se ha realizado correctamente una operación en un dispositivo PKCS#11.
143	El usuario de LDAP no es un usuario válido. (Este programa no soporta LDAP)
144	El usuario de LDAP no es un usuario válido. (Este programa no soporta LDAP)
145	No se ha realizado correctamente la consulta de LDAP. (Este programa no soporta LDAP)
146	Se ha encontrado una cadena de certificados no válida.
147	El certificado de raíz no es de confianza.
148	Se ha encontrado un certificado revocado.

Código de error	Mensaje de error
149	No se ha realizado correctamente la función de objetos de cifrado.
150	No hay disponible ningún origen de datos de lista de revocación de certificados.
151	No hay disponible ninguna señal de cifrado.
152	No está disponible la modalidad FIPS.
153	Hay un conflicto en la configuración de la modalidad FIPS.
154	La contraseña introducida no cumple los requisitos mínimos de seguridad.
200	Se ha producido una anomalía durante la inicialización del programa.
201	La señalización de los argumentos pasado al programa runmqakm no se ha realizado correctamente.
202	El objeto identificado en el mandato no se reconoce.
203	La acción pasada no es una acción -keydb conocida.
204	La acción pasada no es una acción -cert conocida.
205	La acción pasada no es una acción -certreq conocida.
206	Falta una etiqueta en el mandato solicitado.
207	El valor pasado en la etiqueta -version no se reconoce.
208	El valor pasado en la etiqueta -size no se reconoce.
209	El valor pasado en la etiqueta -dn no tiene un formato correcto.
210	El valor pasado en la etiqueta -format no se reconoce.
211	Se ha producido un error al abrir el fichero.
212	PKCS12 no está soportado en esta etapa.
213	La señal de cifrado para la que está intentado cambiar la contraseña no está protegida con contraseña.
214	PKCS12 no está soportado en esta etapa.
215	La contraseña introducida no cumple los requisitos mínimos de seguridad.
216	No está disponible la modalidad FIPS.
217	El número de días que ha introducido para la fecha de caducidad está fuera del intervalo permitido.

Código de error	Mensaje de error
218	La seguridad de la contraseña no cumple los requisitos mínimos.
219	No se ha encontrado certificado predeterminado en la base de datos de claves solicitada.
220	Se ha encontrado un estado de confianza no válido.
221	Se ha encontrado un algoritmo de firma que no está soportado. En esta etapa sólo se soporta MD5 y SHA1.
222	PCKS11 no está soportado para esta operación.
223	La acción pasada no es una acción -random conocida.
224	No se permite una longitud inferior a cero.
225	La longitud mínima de caracteres de la etiqueta -strong es 14.
226	La longitud máxima de caracteres de la etiqueta -strong es 300.
227	El algoritmo MD5 no está soportado en la modalidad FIPS.
228	La etiqueta site no está soportada en el mandato -cert -list. Este atributo se añade por razones de compatibilidad en sentido inverso o mejora futura de potencial.
229	No se reconoce el valor asociado a la etiqueta -ca . El valor debe ser 'true' o 'false'.
230	El valor pasado en la etiqueta -type no es válido.
231	El valor pasado en la etiqueta -expire está por debajo del intervalo permitido.
232	No se soporta el algoritmo de cifrado utilizado o solicitado.
233	Ya existe el destino.

Protección de los detalles de autenticación de base de datos

Si está utilizando la autenticación de nombre de usuario y contraseña para conectarse al gestor de base de datos, puede almacenarlos en el almacén de credenciales XA de MQ para evitar almacenar la contraseña en texto sin formato en el archivo `qm.ini`.

Actualizar XAOpenString para el gestor de recursos

Para utilizar el almacén de credenciales debe modificar XAOpenString en el archivo `qm.ini`. La serie se utiliza para conectarse al gestor de base de datos. Especifique campos sustituibles para identificar donde el nombre de usuario y la contraseña se sustituyen dentro de la serie XAOpenString.

- El campo +USER+ se sustituye por el valor de nombre de usuario almacenado en el almacén XACredentials.
- El campo +PASSWORD+ se sustituye por el valor de contraseña almacenado en el almacén XACredentials.

Los siguientes ejemplos muestran cómo modificar una XAOpenString para que utilice el archivo de credenciales para conectarse a la base de datos.

Conexión con una base de datos Db2

```
XAResourceManager:
  Name=mydb2
  SwitchFile=db2swit
  XAOpenString=db=mydbname,uid=+USER+,pwd=+PASSWORD+,toc=t
  ThreadOfControl=THREAD
```

Conexión a una base de datos Oracle

```
XAResourceManager:
  Name=myoracle
  SwitchFile=oraswit
  XAOpenString=Oracle_XA+Acc=P/+USER+/+PASSWORD++SesTm=35
            +LogDir=/tmp+threads=true
  ThreadOfControl=THREAD
```

Trabajar con las credenciales para la base de datos para el almacén de credenciales XA de MQ

Después de actualizar el archivo `qm.ini` con las series de credenciales sustituibles, debe añadir el nombre de usuario y la contraseña al almacén de credenciales de MQ utilizando el mandato **setmqxacred** . También puede utilizar **setmqxacred** para modificar credenciales existentes, suprimir credenciales o listar credenciales. Los siguientes ejemplos proporcionan algunos casos de uso típicos:

Adición de credenciales

El siguiente mandato guarda de forma segura el nombre de usuario y la contraseña para el gestor de colas QM1 para el recurso mqdb2.

```
setmqxacred -m QM1 -x mydb2 -u user1 -p Password2
```

Actualización de credenciales

Para actualizar el nombre de usuario y la contraseña que se utilizan para conectarse a una base de datos, vuelva a emitir el mandato **setmqxacred** con el nombre de usuario y la contraseña nuevos.

```
setmqxacred -m QM1 -x mydb2 -u user3 -p Password4
```

Debe reiniciar el gestor de colas para que los cambios entren en vigor.

Supresión de credenciales

El siguiente mandato suprime las credenciales:

```
setmqxacred -m QM1 -x mydb2 -d
```

Listado de credenciales

El siguiente mandato lista credenciales:

```
setmqxacred -m QM1 -l
```

Referencia relacionada

setmqxacred

Protección de Managed File Transfer

Directamente tras la instalación y sin ninguna modificación, Managed File Transfer tiene un nivel de seguridad que puede ser adecuado para realizar pruebas o evaluaciones en un entorno protegido. Sin embargo, en un entorno de producción, debe considerar la posibilidad de controlar de manera apropiada quién puede iniciar operaciones de transferencia de archivos, quién puede leer y grabar los archivos que se están transfiriendo y cómo proteger la integridad de los archivos.

Tareas relacionadas

Gestión de autorizaciones de grupo para recursos específicos de MFT

Gestión de autorizaciones para recursos específicos de MFT

"Utilización de Advanced Message Security con Managed File Transfer" en la página 612

Este caso de ejemplo explica cómo configurar Advanced Message Security para proporcionar privacidad de mensajes para los datos que se envían a través de Managed File Transfer.

Referencia relacionada

Autorizaciones para el acceso de MFT a los sistemas de archivos

Propiedad commandPath de MFT

Autorización para publicar mensajes de registro y estado de agentes MFT

Autenticación de conexión de MFT y IBM MQ

La autenticación de conexión permite que un gestor de colas se configure para autenticar aplicaciones utilizando un ID de usuario y una contraseña proporcionados. Si el gestor de colas asociado tiene la seguridad habilitada y requiere detalles de credenciales (ID de usuario y contraseña), la característica de autenticación de conexión debe estar habilitada antes de que se pueda realizar una conexión correcta a un gestor de colas. La autenticación de conexión se puede ejecutar en modalidad de compatibilidad o en modalidad de autenticación MQCSP.

Métodos para suministrar detalles de credenciales

Muchos mandatos de Managed File Transfer dan soporte a los métodos siguientes para suministrar detalles de credenciales:

Detalles proporcionados por los argumentos de línea de mandatos.
Los detalles de las credenciales se pueden especificar utilizando los parámetros **-mquserid** y **-mqpassword**. Si no se proporciona **-mqpassword**, se solicita al usuario la contraseña en la que no se visualiza la entrada.

Los detalles proporcionados desde un archivo de credenciales: `MQMFTCredentials.xml`.
Los detalles de credenciales pueden predefinirse en un archivo `MQMFTCredentials.xml` como texto simple o texto enmascarado.

Para obtener información sobre cómo configurar un archivo MQMFTCredentials.xml en IBM MQ for Multiplatforms , consulte "Configuración de MQMFTCredentials.xml en multiplataformas" en la página 551.

Para obtener información sobre cómo configurar un archivo MQMFTCredentials.xml en IBM MQ for z/OS , consulte "Configuración de MQMFTCredentials.xml en z/OS" en la página 552.

Prioridad

La prioridad de determinar los detalles de credenciales es:

1. Argumento de línea de mandatos.
2. Índice de `MQMFTCredentials.xml` por gestor de colas asociado y usuario que ejecuta el mandato.
3. Índice de `MQMFTCredentials.xml` por gestor de colas asociado.
4. Modalidad de compatibilidad con versiones anteriores predeterminada donde no se proporcionan detalles de credenciales para permitir la compatibilidad con releases anteriores de IBM MQo IBM WebSphere MQ

Notas:

- Los mandatos **fteStartAgent** y **fteStartLogger** no dan soporte al argumento de línea de mandatos **-mquserid**, ni **-mqpassword**, y los detalles de credenciales sólo pueden especificarse con el archivo MQMFTCredentials.xml.
- z/OS

  En z/OS, la contraseña debe ir en mayúsculas, aunque la contraseña del usuario tenga minúsculas. Por ejemplo, si la contraseña del usuario era "password", deberá especificarse como "PASSWORD".

Referencia relacionada

Qué mandato de MFT se conecta a qué gestor de colas

Formato del archivo de credenciales de MFT

Configuración de MQMFTCredentials.xml en multiplataformas

Si Managed File Transfer (MFT) está configurado con la seguridad habilitada, la autenticación de conexión requiere que todos los mandatos de MFT que se conectan con un gestor de colas proporcionen credenciales de ID de usuario y contraseña. De forma similar, los registradores de MFT pueden ser necesarios para especificar un ID de usuario y una contraseña al conectarse a una base de datos. Esta información de credenciales se puede almacenar en el archivo de credenciales MFT .

Acerca de esta tarea

Los elementos del archivo MQMFTCredentials.xml deben ajustarse al esquema MQMFTCredentials.xsd. Para obtener información sobre el formato de MQMFTCredentials.xml, consulte Formato de archivo de credenciales MFT.

Puede encontrar un archivo de credenciales de ejemplo en el directorio MQ_INSTALLATION_PATH/mqft/samples/credentials .

Puede tener un archivo de credenciales de MFT para el gestor de colas de coordinación, uno para el gestor de colas de mandatos, uno para cada agente y uno para cada registrador. De forma alternativa, puede tener un archivo que sea utilizado por todo en la topología.

La ubicación predeterminada del archivo de credenciales MFT es la siguiente:

Linux UNIX **UNIX and Linux**

$HOME

Windows **Windows**

%USERPROFILE% o %HOMEDRIVE%%HOMEPATH%

Si el archivo de credenciales se almacena en una ubicación diferente, puede utilizar las propiedades siguientes para especificar dónde deben buscarlo los mandatos:

Tabla 91. : Propiedades que definen la ubicación del archivo MQMFTCredentials.xml para varios mandatos.

Tipo de mandato	Archivo de propiedades	Nombre de propiedad
Mandato que se conecta al gestor de colas de coordinación	coordination.properties	coordinationQMgrAuthenticationCredentialsFile
Mandato que se conecta al gestor de colas de mandatos	connection.properties	connectionQMgrAuthenticationCredentialsFile
Mandato que se conecta a un proceso de agente	agent.properties	agentQMgrAuthenticationCredentialsFile

Tabla 91. : Propiedades que definen la ubicación del archivo MQMFTCredentials.xml para varios mandatos. (continuación)		
Tipo de mandato	**Archivo de propiedades**	**Nombre de propiedad**
Mandato que se conecta a un proceso de registrador	logger.properties	loggerQMgrAuthenticationCredentialsFile

Tabla 92. : Propiedades que definen la ubicación del archivo MQMFTCredentials.xml para agentes y procesos de registrador.		
Tipo de mandato	**Archivo de propiedades**	**Nombre de propiedad**
Agentes de MFT	agent.properties	agentQMgrAuthenticationCredentialsFile
MFT registradores	logger.properties	loggerQMgrAuthenticationCredentialsFile

Para obtener detalles sobre qué mandatos y procesos se conectan a qué gestor de colas, consulte Qué mandatos y procesos de MFT se conectan a qué gestor de colas.

Puesto que el archivo de credenciales contiene información de ID de usuario y contraseña, requiere permisos especiales para impedir el acceso no autorizado al mismo:

Linux UNIX **UNIX and Linux**

```
chown <agent owner userid>
chmod 600
```

Windows **Windows**

Asegúrese de que la herencia no está habilitada y, a continuación, elimine todos los ID de usuario excepto los que ejecuten el agente o registrador que utilizarán el archivo de credenciales.

Los detalles de credenciales utilizados para conectarse a un gestor de colas de coordinación de MFT , en el plug-in de IBM MQ Explorer Managed File Transfer para, depende del tipo de configuración:

Global (configuración en disco local)

Una configuración global utiliza el archivo de credenciales especificado en las propiedades de coordinación y mandatos.

Local (definida dentro de IBM MQ Explorer):

Una configuración local utiliza las propiedades de los detalles de conexión del gestor de colas asociado en IBM MQ Explorer.

Tareas relacionadas

“Habilitación de la autenticación de conexión para MFT” en la página 554

La autenticación de conexión del plugin de IBM MQ Explorer MFT que se conecta con un gestor de colas de coordinación o un gestor de colas de mandatos, y la autenticación de conexión para un agente de Managed File Transfer que se conecta con un gestor de colas de coordinación o un gestor de colas de mandatos se puede ejecutar en modalidad de compatibilidad o en modalidad de autenticación MQCSP.

Referencia relacionada

Formato del archivo de credenciales de MFT

fteObfuscate: cifrar datos confidenciales

z/OS Configuración de MQMFTCredentials.xml en z/OS

Si Managed File Transfer (MFT) está configurado con la seguridad habilitada, la autenticación de conexión requiere que todos los agentes de MFT y los mandatos que se conectan a un gestor de colas proporcionen credenciales de ID de usuario y contraseña.

De forma similar, los registradores de MFT pueden ser necesarios para especificar un ID de usuario y una contraseña al conectarse a una base de datos.

Esta información de credenciales se puede almacenar en el archivo de credenciales MFT . Tenga en cuenta que los archivos de credenciales son opcionales, sin embargo, es más fácil definir el archivo o archivos que necesita antes de personalizar el entorno.

Además de esto, si tiene archivos de credenciales, recibirá menos mensajes de aviso. Los mensajes de aviso le informan de que MFT considera que la seguridad del gestor de colas está desactivada y, por lo tanto, no está suministrando los detalles de autenticación.

Puede encontrar un archivo de credenciales de ejemplo en el directorio `MQ_INSTALLATION_PATH/mqft/samples/credentials` .

A continuación se muestra un ejemplo de un archivo `MQMFTCredentials.xml`:

```
<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MFTCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MFTCredentials.xsd">
  <tns:qmgr name="MQPH" user="ADMIN" mqUserId="JOHNDOEH" mqPassword="cXXXX" />
  <tns:qmgr name="MQPI" user="ADMIN" mqUserId="JOHNDOEI" mqPassword="yXXXX" />
  <tns:qmgr name="MQPH" mqUserId="NONEH" mqPassword="yXXXX" />
  <tns:qmgr name="MQPI" mqUserId="NONEI" mqPassword="yXXXX" />
</tns:mqmftCredentials>
```

Cuando un trabajo con el ID de usuario ADMIN se ha conectar al gestor de colas MQPH, pasa el ID de usuario *JOHNDOEH* y utiliza la contraseña *cXXXX*.

Si el trabajo se ejecuta con otro ID de usuario, y conecta MQPH, dicho trabajo pasa el ID de usuario *NONEH* y la contraseña *yXXXX*.

La ubicación predeterminada del archivo `MQMFTCredentials.xml` es el directorio de inicio del usuario en z/OS UNIX System Services (USS). También es posible almacenar el archivo en una ubicación diferente en USS o en un miembro dentro de un conjunto de datos particionados.

Si el archivo de credenciales se almacena en una ubicación diferente, puede utilizar las propiedades siguientes para especificar dónde deben buscarlo los mandatos:

Tabla 93. : Propiedades que definen la ubicación del archivo MQMFTCredentials.xml para varios mandatos.

Tipo de mandato	Archivo de propiedades	Nombre de propiedad
Mandato que se conecta al gestor de colas de coordinación	coordination.properties	coordinationQMgrAuthenticationCredentialsFile
Mandato que se conecta al gestor de colas de mandatos	connection.properties	connectionQMgrAuthenticationCredentialsFile
Mandato que se conecta a un proceso de agente	agent.properties	agentQMgrAuthenticationCredentialsFile
Mandato que se conecta a un proceso de registrador	logger.properties	loggerQMgrAuthenticationCredentialsFile

Tabla 94. : Propiedades que definen la ubicación del archivo MQMFTCredentials.xml para agentes y procesos de registrador.

Tipo de mandato	Archivo de propiedades	Nombre de propiedad
Agentes de MFT	agent.properties	agentQMgrAuthenticationCredentialsFile
MFT registradores	logger.properties	loggerQMgrAuthenticationCredentialsFile

Para obtener detalles sobre qué mandatos y procesos se conectan a qué gestor de colas, consulte Qué mandatos y procesos de MFT se conectan a qué gestor de colas.

Para crear el archivo de credenciales dentro de un conjunto de datos particionados, realice los pasos siguientes:

- Cree un PDSE con el formato VB y la longitud de registro lógico (Lrecl) de 200.
- Cree un miembro dentro del conjunto de datos, anote el conjunto de datos y el miembro y añada el código siguiente al miembro:

```
<?xml version="1.0" encoding="IBM-1047"?>
<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MQMFTCredentials"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://wmqfte.ibm.com/MFTCredentials MQMFTCredentials.xsd">
  <!—-credentials information goes here-->
</tns:mqmftCredentials>
```

Puede proteger el archivo de credenciales utilizando un producto de seguridad, por ejemplo, RACF, pero los ID de usuario que ejecutan los mandatos Managed File Transfer y que administran los procesos de agente y registrador, necesitan acceso de lectura a este archivo.

Puede ocultar la información de este archivo utilizando el JCL del miembro BFGCROBS. Éste toma el archivo y cifra el ID de usuario y la contraseña de IBM MQ. Por ejemplo, el miembro BFGCROBS tendrá la línea

```
<tns:qmgr name="MQPI" user="JOHNDOE2" mqUserId="JOHNDOE1" mqPassword="yXXXX" />
```

y creará

```
<tns:qmgr mqPasswordCipher="e977c61e9b9c363c" mqUserIdCipher="c394c5887867157c"
name="MQPI" user="JOHNDOE2"/>
```

Si desea conservar la correlación del ID de usuario con el ID de usuario de IBM MQ, puede añadir comentarios al archivo. Por ejemplo

```
<!--      name="MQPI" user="ADMIN"    mqUserId="JOHNDOE1 -->
```

Estos comentarios no los modifica el proceso de ocultación.

Tenga en cuenta que el contenido se oculta pero no se utiliza un cifrado fuerte. Debe limitar los ID de usuario con acceso al archivo.

Tareas relacionadas

“Configuración de MQMFTCredentials.xml en multiplataformas” en la página 551
Si Managed File Transfer (MFT) está configurado con la seguridad habilitada, la autenticación de conexión requiere que todos los mandatos de MFT que se conectan con un gestor de colas proporcionen credenciales de ID de usuario y contraseña. De forma similar, los registradores de MFT pueden ser necesarios para especificar un ID de usuario y una contraseña al conectarse a una base de datos. Esta información de credenciales se puede almacenar en el archivo de credenciales MFT .

Habilitación de la autenticación de conexión para MFT

La autenticación de conexión del plugin de IBM MQ Explorer MFT que se conecta con un gestor de colas de coordinación o un gestor de colas de mandatos, y la autenticación de conexión para un agente de Managed File Transfer que se conecta con un gestor de colas de coordinación o un gestor de colas de mandatos se puede ejecutar en modalidad de compatibilidad o en modalidad de autenticación MQCSP.

Acerca de esta tarea

Antes de IBM MQ 9.1.1, la modalidad de compatibilidad es el valor predeterminado para la autenticación de conexión. Sin embargo, puede inhabilitar la modalidad de compatibilidad predeterminada y habilitar la modalidad de autenticación MQCSP.

V 9.1.1 A partir de IBM MQ 9.1.1, el modo de autenticación MQCSP es el modo predeterminado.

Para la autenticación de conexión para el plug-in de IBM MQ Explorer Managed File Transfer o para los agentes de Managed File Transfer que se conectan a un gestor de colas utilizando el transporte CLIENT, las contraseñas de más de 12 caracteres sólo están soportadas para la modalidad de autenticación MQCSP. Si especifica una contraseña de más de 12 caracteres de longitud cuando se autoriza el uso de la modalidad de compatibilidad, se produce un error y el agente no se autentica con el gestor de colas. Consulte el mensaje BFGAG0187E en la sección Mensajes de diagnóstico: BFGAG0001 - BFGAG9999.

Procedimiento

- Para seleccionar la modalidad de autenticación de conexión para un gestor de colas de coordinación o un gestor de colas de mandatos en IBM MQ Explorer, complete los pasos siguientes:
  a) Seleccione el gestor de colas al que desea conectarse.
  b) Pulse con el botón derecho del ratón y seleccione **Detalles de conexión -> Propiedades** en el menú emergente.
  c) Pulse la pestaña **ID de usuario**.
  d) Asegúrese de que el recuadro de selección para la modalidad de autenticación de conexión que desea utilizar esté seleccionado:
     - V 9.1.0 A partir de IBM MQ 9.1.0, de forma predeterminada, el recuadro de selección **Modalidad de compatibilidad de identificación de usuario** no está seleccionado. Esto significa que si se selecciona el recuadro de selección **Habilitar identificación de usuario**, IBM MQ Explorer utilizará la autenticación MQCSP al conectarse al gestor de colas. Si IBM MQ Explorer debe conectarse al gestor de colas utilizando la modalidad de compatibilidad en lugar de la autenticación MQCSP, asegúrese de que estén seleccionados los recuadros de selección **Habilitar identificación de usuario** y **Modalidad de compatibilidad de identificación de usuario**.
     - Antes de IBM MQ 9.1.0, de forma predeterminada, el recuadro de selección **Modalidad de compatibilidad de identificación de usuario** está seleccionado. Esto significa que si se selecciona el recuadro de selección **Habilitar identificación de usuario**, IBM MQ Explorer utilizará la modalidad de compatibilidad al conectar con el gestor de colas. Si IBM MQ Explorer debe conectarse al gestor de colas utilizando la autenticación MQCSP, asegúrese de que esté seleccionado el recuadro de selección **Habilitar identificación de usuario** y que no esté seleccionado **Modalidad de compatibilidad de identificación de usuario**.
- Para habilitar o inhabilitar la modalidad de autenticación MQCSP para un agente de Managed File Transfer utilizando el archivo `MQMFTCredentials.xml`, añada el parámetro **useMQCSPAuthentication** al archivo `MQMFTCredentials.xml` para el usuario pertinente.

  El parámetro **useMQCSPAuthentication** tiene los valores siguientes:

  true
  La modalidad de autenticación MQCSP se utiliza para autenticar el usuario con el gestor de colas.

  V 9.1.1 A partir de IBM MQ 9.1.1, `true` es el valor predeterminado. Si no se especifica el parámetro **useMQCSPAuthentication**, se establece de forma predeterminada en `true` y la modalidad de autenticación MQCSP se utiliza para autenticar el usuario con el gestor de colas.

  falso
  La modalidad de compatibilidad se utiliza para autenticar el usuario con el gestor de colas.

  Antes de IBM MQ 9.1.1, si no se especifica el parámetro **useMQCSPAuthentication**, se establece de forma predeterminada en `false` y se utiliza la modalidad de compatibilidad para autenticar el usuario con el gestor de colas.

En el ejemplo siguiente se muestra cómo establecer el parámetro **useMQCSPAuthentication** en el archivo `MQMFTCredentials.xml`:

```
<tns:qmgr name="CoordQueueMgr" user="ernest" mqUserId="ernest"
    mqPassword="AveryL0ngPassw0rd2135" useMQCSPAuthentication="true"/>
```

Conceptos relacionados

"Protección por contraseña MQCSP" en la página 30
A partir de la IBM MQ 8.0, puede enviar contraseñas incluidas en la estructura MQCSP tanto si se han protegido mediante la función IBM MQ como si se han cifrado mediante el cifrado TLS.

Referencia relacionada

"Autenticación de conexión de MFT y IBM MQ" en la página 550
La autenticación de conexión permite que un gestor de colas se configure para autenticar aplicaciones utilizando un ID de usuario y una contraseña proporcionados. Si el gestor de colas asociado tiene la seguridad habilitada y requiere detalles de credenciales (ID de usuario y contraseña), la característica de autenticación de conexión debe estar habilitada antes de que se pueda realizar una conexión correcta a un gestor de colas. La autenticación de conexión se puede ejecutar en modalidad de compatibilidad o en modalidad de autenticación MQCSP.

Formato del archivo de credenciales de MFT

Recintos de seguridad de MFT

Puede restringir el área del sistema de archivos a la que puede acceder el agente como parte de una transferencia. El área a la que el agente está restringida se denomina el recinto de seguridad. Puede aplicar restricciones al agente o al usuario que solicita una transferencia.

Los recintos de seguridad no están soportados cuando el agente es un agente de puente de protocolo o un agente de puente Connect:Direct. No puede utilizar recintos de seguridad de agente para agentes que tienen que transferir a o desde colas de IBM MQ.

Referencia relacionada

"Trabajo con recintos de seguridad de agente MFT" en la página 556
Para añadir un nivel adicional de seguridad a Managed File Transfer, puede restringir el área de un sistema de archivos a la que un agente puede acceder.

"Trabajo con recintos de seguridad de usuario de MFT" en la página 558
Puede restringir el área del sistema de archivos de y a la que transferir los archivos dependiendo del nombre de usuario de MQMD que solicita la transferencia.

Trabajo con recintos de seguridad de agente MFT

Para añadir un nivel adicional de seguridad a Managed File Transfer, puede restringir el área de un sistema de archivos a la que un agente puede acceder.

No puede utilizar recintos de seguridad de agente para agentes que transfieren a o desde colas de IBM MQ. En su lugar, se puede implementar la restricción de acceso a las colas de IBM MQ con recintos de seguridad utilizando el recinto de seguridad de usuario que es la solución recomendada para los requisitos de recinto de seguridad. Para obtener más información sobre el recinto de seguridad de usuario, consulte "Trabajo con recintos de seguridad de usuario de MFT" en la página 558

Para habilitar el recinto de seguridad de agente, añada la siguiente propiedad al archivo `agent.properties` del agente que desea restringir:

```
sandboxRoot=[!]restricted_directory_nameseparator...separator[!]restricted_directory_name
```

donde:

- *restricted_directory_name* es una vía de acceso de directorio que se debe permitir o denegar.

- ! es opcional y especifica que se deniega (excluye) el siguiente valor para *restricted_directory_name* . Si no se especifica ! , *restricted_directory_name* es una vía de acceso permitida (incluida).
- *separator* es el separador específico de la plataforma.

Por ejemplo, si sólo desea restringir el acceso que AGENT1 tiene al directorio `/tmp`, pero no desea permitir el acceso al subdirectorio `private`, establezca la propiedad de la siguiente manera en el archivo `agent.properties` perteneciente a AGENT1: `sandboxRoot=/tmp:!/tmp/private`.

La propiedad sandboxRoot se describe en Propiedades avanzadas del agente.

No se soporta la creación de recintos de seguridad de usuario y de agente en los agentes de puente de protocolo o en agentes de puente de Connect:Direct.

Trabajar en un recinto de seguridad en plataformas UNIX, Linux y Windows

ULW En las plataformas UNIX, Linux y Windows, el recinto de seguridad restringe los directorios en los que puede leer y escribir un Managed File Transfer Agent. Cuando se activa el recinto de seguridad, el Managed File Transfer Agent puede leer y escribir en los directorios en los que se permite y en sus subdirectorios a menos que estos estén especificados como denegados en sandboxRoot. El recinto de seguridad de Managed File Transfer no tiene prioridad sobre la seguridad del sistema operativo. El usuario que ha iniciado el Managed File Transfer Agent debe tener el acceso de nivel de sistema operativo adecuado a cualquier directorio para poder leer o escribir en dicho directorio. No se sigue un enlace simbólico a un directorio si el directorio enlazado está fuera de los directorios (y subdirectorios) sandboxRoot.

Trabajar en un recinto de seguridad en z/OS

z/OS En z/OS, el recinto de seguridad restringe los cualificadores de nombre de conjunto de datos en los que el Managed File Transfer Agent puede leer y escribir. El usuario que ha iniciado el Managed File Transfer Agent debe tener las correspondientes autorizaciones del sistema operativo para los conjuntos de datos implicados. Si escribe un valor de calificador de nombre de conjunto de datos sandboxRoot entre comillas dobles, el valor sigue el convenio normal de z/OS y se trata como un valor totalmente calificado. Si omite las comillas dobles, al directorio sandboxRoot se le antepone como prefijo el ID de usuario actual. Por ejemplo, si establece la propiedad sandboxRoot en lo siguiente: `sandboxRoot=//test`, el agente puede acceder a los siguientes conjuntos de datos (en notación z/OS estándar) `//`*username*`.test.**` En tiempo de ejecución, si los niveles iniciales del nombre de conjunto de datos totalmente resuelto no coinciden con el sandboxRoot, se rechaza la solicitud de transferencia.

Trabajar en un recinto de seguridad en sistemas IBM i

IBM i En los archivos del sistema de archivos integrado en sistemas IBM i, el recinto de seguridad restringe los directorios en los que un Managed File Transfer Agent puede leer y escribir. Cuando se activa el recinto de seguridad, el Managed File Transfer Agent puede leer y escribir en los directorios en los que se permite y en sus subdirectorios a menos que estos estén especificados como denegados en sandboxRoot. El recinto de seguridad de Managed File Transfer no tiene prioridad sobre la seguridad del sistema operativo. El usuario que ha iniciado el Managed File Transfer Agent debe tener el acceso de nivel de sistema operativo adecuado a cualquier directorio para poder leer o escribir en dicho directorio. No se sigue un enlace simbólico a un directorio si el directorio enlazado está fuera de los directorios (y subdirectorios) sandboxRoot.

Referencia relacionada

“Comprobaciones adicionales de transferencias de comodín” en la página 561
Si se ha configurado un agente con un recinto de pruebas de usuario o agente para poder restringir las ubicaciones en las que el agente puede transferir archivos, y puede especificar que comprobaciones adicionales se van a realizar en transferencias de comodín para dicho agente.

“Trabajo con recintos de seguridad de agente MFT” en la página 556
Para añadir un nivel adicional de seguridad a Managed File Transfer, puede restringir el área de un sistema de archivos a la que un agente puede acceder.

El archivo MFT agent.properties

Trabajo con recintos de seguridad de usuario de MFT

Puede restringir el área del sistema de archivos de y a la que transferir los archivos dependiendo del nombre de usuario de MQMD que solicita la transferencia.

Los recintos de seguridad de usuario no están soportados cuando el agente es un agente de puente de protocolo o un agente de puente Connect:Direct.

Para habilitar los recintos de seguridad, añada la siguiente propiedad al archivo agent.properties para el agente que desea restringir:

```
userSandboxes=true
```

Cuando esta propiedad está presente y se establece en true, el agente utiliza la información del archivo *MQ_DATA_PATH*/mqft/config/*coordination_qmgr_name*/agents/*agent_name*/UserSandboxes.xml para determinar a qué partes del sistema de archivos puede acceder el usuario que solicita la transferencia.

El XML UserSandboxes.xml se compone de un elemento <agent> que contiene cero o más elementos <sandbox> . Estos elementos describen qué reglas se aplican a qué usuarios. El atributo user del elemento <sandbox> es un patrón que se utiliza para buscar coincidencias con el usuario MQMD de la solicitud.

El agente vuelve a cargar periódicamente el archivo UserSandboxes.xml y cualquier cambio válido en el archivo afectará al comportamiento del agente. El intervalo de recarga predeterminado es de 30 segundos. Este intervalo se puede cambiar especificando la propiedad de agente xmlConfigReloadInterval en el archivo agent.properties.

Si se especifica el atributo o valor userPattern="regex", el atributo user se interpreta como una expresión regular Java. Si desea más información, consulte Expresiones regulares utilizadas por MFT.

Si no especifica el atributo o valor userPattern="regex", el atributo user se interpreta como un patrón con los siguientes caracteres comodín:

- asterisco (*), que representa cero o más caracteres
- signo de interrogación (?), que representa exactamente un carácter

Las coincidencias se realizan en el orden en el que los elementos <sandbox> se listan en el archivo. Sólo se utiliza la primera coincidencia, todas las siguientes coincidencias potenciales en el archivo se ignoran. Si ninguno de los elementos <sandbox> especificados en el archivo coincide con el usuario MQMD asociado con el mensaje de solicitud de transferencia, la transferencia no puede acceder al sistema de archivos. Cuando se encuentra una coincidencia entre el nombre de usuario MQMD y un atributo user, la coincidencia identifica un conjunto de reglas dentro de un elemento <sandbox> que se aplican a la transferencia. Este conjunto de reglas se utiliza para determinar qué archivos, o conjuntos de datos, pueden leerse o escribirse como parte de la transferencia.

Cada conjunto de reglas puede especificar un elemento <read> , que identifica qué archivos se pueden leer, y un elemento <write> que identifica qué archivos se pueden escribir. Si omite los elementos <read> o <write> de un conjunto de reglas, se supone que el usuario asociado con dicho conjunto de reglas no tiene permiso para realizar ninguna lectura ni escritura, según corresponda.

Nota: El elemento <read> debe estar antes del elemento <write>, y el elemento <include> debe ser anterior al elemento <exclude>, en el archivo UserSandboxes.xml.

Cada elemento <read> o <write> contiene uno o más patrones que se utilizan para determinar si un archivo está en el recinto de seguridad y se puede transferir. Especifique estos patrones utilizando los elementos <include> y <exclude>. El atributo name del elemento <include> o <exclude> especifica el patrón que debe coincidir. Un atributo type opcional especifica si el valor de nombre es un

patrón de cola o un archivo. Si no se especifica el atributo `type`, el agente trata el patrón como un patrón de vía de acceso de archivo o de directorio. Por ejemplo:

```
<tns:read>
    <tns:include name="/home/user/**"/>
    <tns:include name="USER.**" type="queue"/>
    <tns:exclude name="/home/user/private/**"/>
</tns:read>
```

El agente utiliza los patrones `<include>` y `<exclude>` `name` para determinar si los archivos, conjuntos de datos o colas se pueden leer o grabar en ellos. Se permite una operación si el nombre canónico de la vía de acceso de archivo, del conjunto de datos o de la cola coincide con al menos uno de los patrones incluidos y exactamente cero de los patrones excluidos. Los patrones especificados utilizando el atributo `name` de los elementos `<include>` y `<exclude>` utilizan los convenios y separadores de vía de acceso correspondientes a la plataforma en que se está ejecutando el agente. Si especifica vías de acceso relativas, las vías de acceso serán resueltas en la propiedad `transferRoot` del agente.

Cuando se especifica una restricción de cola, una sintaxis de QUEUE@QUEUEMANAGER está soportada, con las reglas siguientes:

- Si falta el carácter de arroba (@) en la entrada, el patrón se trata como un nombre de cola al que se puede acceder a cualquier gestor de colas. Por ejemplo, si el patrón es `name`, se trata de la misma manera que `name@**`.
- Si el carácter de arroba (@) es el primer carácter de la entrada, el patrón se trata como un nombre de gestor de colas y se puede acceder a todas las colas del gestor de colas. Por ejemplo, si el patrón es `@name`, se trata de la misma manera que `**@name`..

Los siguientes caracteres comodín tienen un significado especial cuando se especifican como parte del atributo `name` de los elementos `<include>` y `<exclude>`:

*****
: Un único asterisco coincide con cero o más caracteres en un nombre de directorio, o en un calificador de un nombre de conjunto de datos o nombre de cola .

?
: Un signo de interrogación coincide exactamente con un carácter en un nombre de directorio, o en un calificador de un nombre de conjunto de datos o nombre de cola.

******
: Dos caracteres de asterisco coinciden con cero o más nombres de directorio, o cero o más calificadores en un nombre de conjunto de datos de o nombre de cola de . Además las vías de acceso que finalizan con un separador de vía de acceso tienen dos asteriscos "**" implícitos añadidos al final de la vía de acceso. Por lo tanto, `/home/user/` es el mismo que `/home/user/**`.

Por ejemplo:

- `/**/test/**` coincide con cualquier archivo con un directorio `test` en su vía de acceso
- `/test/file?` coincide con cualquier archivo del directorio `/test` que empiece por la serie `file` seguido de cualquier carácter único
- `c:\test\*.txt` coincide con cualquier archivo dentro del directorio `c:\test` con una extensión `.txt`
- `c:\test\**\*.txt` coincide con cualquier archivo dentro del directorio `'c:\test` o uno de sus subdirectorios con una extensión `.txt`
- z/OS `//'TEST.*.DATA'` coincide con cualquier conjunto de datos que tenga el primer calificador TEST, tiene cualquier segundo calificador y un tercero de DATA.
- *@QM1 coincide con cualquier cola del gestor de colas QM1 que tenga un único calificador.
- TEST.*.QUEUE@QM1 coincide con cualquier cola del gestor de colas QM1 que tiene el primer cualificador de TEST, tiene cualquier segundo cualificador y un tercero de QUEUE.
- **@QM1 coincide con cualquier cola del gestor de colas QM1.

Enlaces simbólicos

Debe resolver por completo los enlaces simbólicos que se utilizan en las vías de acceso de archivo en el archivo `UserSandboxes.xml` especificando enlaces fijos en los elementos `<include>` y `<exclude>`. Por ejemplo, si tiene un enlace simbólico donde `/var` se correlaciona con `/SYSTEM/var`, debe especificar esta vía de acceso como `<tns:include name="/SYSTEM/var"/>`, de lo contrario la transferencia prevista fallará con un error de seguridad de recinto de seguridad de usuario.

Ejemplo

Este ejemplo muestra cómo permitir que el usuario con el nombre de usuario MQMD `guest` transfiera cualquier archivo desde el directorio `/home/user/public` o cualquiera de sus subdirectorios en el sistema donde se ejecuta el agente AGENT_JUPITER, añadiendo el siguiente elemento `<sandbox>` al archivo `UserSandboxes.xml` en el directorio de configuración de AGENT_JUPITER:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
        xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
        xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
    <tns:agent>
        <tns:sandbox user="guest">
            <tns:read>
                <tns:include name="/home/user/public/**"/>
             </tns:read>
        </tns:sandbox>
    </tns:agent>
</tns:userSandboxes>
```

Ejemplo

En este ejemplo se muestra cómo permitir que cualquier usuario con el nombre de usuario MQMD `account` seguido de un único dígito, por ejemplo, `account4`, complete las siguientes acciones:

- Transfiera cualquier archivo desde el directorio `/home/account` o cualquiera de sus subdirectorios, excluyendo el directorio `/home/account/private` en el sistema donde se está ejecutando el agente AGENT_SATURN
- Transfiera cualquier archivo al directorio `/home/account/output` o a cualquiera de sus subdirectorios en el sistema donde se está ejecutando AGENT_SATURN
- Leer mensajes de las colas del gestor de colas local que empiezan por el prefijo `ACCOUNT.` a menos que empiece por `ACCOUNT.PRIVATE.` (es decir, que tenga `PRIVATE` en el segundo nivel).
- Transfiera datos a las colas que empiezan con el prefijo `ACCOUNT.OUTPUT.` en cualquier gestor de colas.

Para permitir que un usuario con el nombre de usuario MQMD `account` complete estas acciones, añada el siguiente elemento `<sandbox>` al archivo `UserSandboxes.xml`, en el directorio de configuración de AGENT_SATURN:

```
<?xml version="1.0" encoding="UTF-8"?>
<tns:userSandboxes
            xmlns:tns="http://wmqfte.ibm.com/UserSandboxes"
            xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="http://wmqfte.ibm.com/UserSandboxes UserSandboxes.xsd">
    <tns:agent>
        <tns:sandbox user="account[0-9]" userPattern="regex">
            <tns:read>
                <tns:include name="/home/account/**"/>
                <tns:include name="ACCOUNT.**" type="queue"/>
                <tns:exclude name="ACCOUNT.PRIVATE.**" type="queue"/>
                <tns:exclude name="/home/account/private/**"/>
                                                    </tns:read>
            <tns:write>
                <tns:include name="/home/account/output/**"/>
                <tns:include name="ACCOUNT.OUTPUT.**" type="queue"/>
            </tns:write>
        </tns:sandbox>
```

```
    </tns:agent>
</tns:userSandboxes>
```

Referencia relacionada

“Comprobaciones adicionales de transferencias de comodín” en la página 561
Si se ha configurado un agente con un recinto de pruebas de usuario o agente para poder restringir las ubicaciones en las que el agente puede transferir archivos, y puede especificar que comprobaciones adicionales se van a realizar en transferencias de comodín para dicho agente.

El archivo MFT `agent.properties`

Comprobaciones adicionales de transferencias de comodín

Si se ha configurado un agente con un recinto de pruebas de usuario o agente para poder restringir las ubicaciones en las que el agente puede transferir archivos, y puede especificar que comprobaciones adicionales se van a realizar en transferencias de comodín para dicho agente.

Propiedad additionalWildcardSandboxChecking

Para habilitar una comprobación adicional de transferencias de comodín, añada la siguiente propiedad al archivo `agent.properties` para el agente que desea comprobar.

```
additionalWildcardSandboxChecking=true
```

Si esta propiedad está establecida en true y el agente realiza una solicitud de transferencia que intenta leer una ubicación que está fuera del recinto de seguridad definido para la coincidencia de archivos del comodín, la transferencia falla. Si hay varias transferencias dentro de una solicitud de transferencia, y una de estas solicitudes falla debido a que intenta leer una ubicación fuera del recinto de seguridad, toda la transferencia falla. Si la comprobación falla, la razón del fallo se proporciona en un mensaje de error.

Si se omite la propiedad additionalWildcardSandboxChecking property del archivo `agent.properties` de un agente o se establece en false, no se realiza ninguna comprobación adicional en las transferencias de comodín para ese agente.

Mensajes de error para la comprobación de comodín

Los mensajes que se notifican cuando se realiza una solicitud de transferencia de comodín a una ubicación fuera de una ubicación de recinto de seguridad configurada son los siguientes.

Aparece el mensaje siguiente cuando una vía de acceso de archivo de comodín en una solicitud de transferencia se encuentra fuera del recinto de seguridad restringido.

```
BFGSS0077E: Se ha rechazado el intento de leer la vía de acceso de archivo vía de acceso.
La vía de acceso del archivo está situada fuera del recinto de seguridad de transferencias
restringido.
```

Aparece el mensaje siguiente cuando una transferencia dentro de una solicitud de transferencia múltiple contiene una solicitud de transferencia de comodín donde la vía de acceso se encuentra fuera del recinto de seguridad restringido.

```
BFGSS0078E: Se ha hecho caso omiso del intento de leer la vía de acceso de archivo: vía de acceso
como otra transferencia.
El elemento de la transferencia gestionada ha intentado leer fuera del recinto de seguridad de
transferencias restringido.
```

Aparece el mensaje siguiente cuando un archivo se encuentra fuera del recinto de seguridad restringido:

```
BFGSS0079E: Se ha denegado el intento de leer el archivo vía de acceso de archivo.
El archivo está situado fuera del recinto de seguridad de transferencias restringido.
```

Aparece el mensaje siguiente en una solicitud de transferencia múltiple donde otra solicitud de transferencia de comodín ha causado que se hiciera caso omiso de esta:

```
BFGSS0079E: Se ha hecho caso omiso del intento de leer el archivo: vía de acceso de archivo como
otra transferencia.
El elemento de la transferencia gestionada ha intentado leer fuera del recinto de seguridad de
transferencias restringido.
```

En el caso de transferencias de archivos únicas que no incluyen caracteres comodín, el mensaje notificado cuando la transferencia implica que un archivo que se encuentra fuera del recinto de seguridad no se ha modificado en releases anteriores:

```
Errores con BFGIO0056E: se ha denegado el intento de leer el archivo "ARCHIVO".
El archivo está situado fuera del recinto de seguridad de transferencias restringido.
```

Referencia relacionada

"Trabajo con recintos de seguridad de usuario de MFT" en la página 558
Puede restringir el área del sistema de archivos de y a la que transferir los archivos dependiendo del nombre de usuario de MQMD que solicita la transferencia.

"Trabajo con recintos de seguridad de agente MFT" en la página 556
Para añadir un nivel adicional de seguridad a Managed File Transfer, puede restringir el área de un sistema de archivos a la que un agente puede acceder.

El archivo MFT `agent.properties`

Configurar el cifrado SSL o TLS para MFT

Puede utilizar SSL o TLS con IBM MQ Managed File Transfer para proteger la comunicación entre los agentes y sus gestores de colas de agente, los mandatos y los gestores de colas a los que se están conectando, y las diversas conexiones de gestor de colas con el gestor de colas dentro de la topología.

Antes de empezar

Puede utilizar el cifrado SSL o TLS para cifrar los mensajes que fluyen a través de una topología de IBM MQ Managed File Transfer . Incluyen los siguientes:

- Mensajes que pasan entre un agente y su gestor de colas de agente.
- Mensajes para los mandatos y los gestores de colas a los que se están conectando.
- Mensajes internos que fluyen entre los gestores de colas de agente, los gestores de colas de mandatos y el gestor de colas de coordinación dentro de la topología.

Acerca de esta tarea

Si desea información general sobre cómo utilizar SSL con IBM MQ, consulte "Trabajar con SSL/TLS" en la página 277. Desde el punto de vista de IBM MQ, Managed File Transfer es una aplicación de cliente Java estándar.

Siga estos pasos para utilizar SSL con Managed File Transfer:

Procedimiento

1. Cree un archivo de almacén de confianza y, opcionalmente, un archivo de almacén de claves (estos archivos pueden ser el mismo archivo). Si no necesita autenticación de cliente (es decir, SSLCAUTH=OPTIONAL en canales) no necesita proporcionar un almacén de claves. Sólo necesita un almacén de confianza para autenticar el certificado del gestor de colas.

   El algoritmo de clave utilizado para crear certificados para el almacén de confianza y los almacenes de claves debe ser RSA para trabajar con IBM MQ.
2. Configure el gestor de colas de IBM MQ para que utilice SSL.

   Para obtener información sobre cómo configurar un gestor de colas para que utilice SSL mediante IBM MQ Explorer, por ejemplo, consulte Configurar SSL en los gestores de colas.
3. Guarde el archivo de almacén de confianza y el archivo de almacén de claves (si dispone de uno) en una ubicación adecuada. Una ubicación sugerida es el directorio *config_directory*/*coordination_qmgr*/agents/*agent_name* .
4. Establezca las propiedades SSL según sea necesario para cada gestor de colas habilitado para SSL en el archivo de propiedades de Managed File Transfer adecuado. Cada conjunto de propiedades se refiere a un gestor de colas separado (agente, coordinación y mandato) aunque un gestor de colas puede ejecutar dos o más de estos roles.

Se precisa una de las propiedades **CipherSpec** o **CipherSuite**, de lo contrario, el cliente intente conectarse sin SSL. Se suministran las propiedades **CipherSpec** y **CipherSuite** debido a las diferencias terminológicas entre IBM MQ y Java. Managed File Transfer acepta la propiedad y no efectúa la conversión necesaria, para que no necesite establecer ambas propiedades. Si especifica las propiedades **CipherSpec** o **CipherSuite**, **CipherSpec** tiene prioridad.

El parámetro **PeerName** es opcional. Puede establecer la propiedad en el nombre distinguido del gestor de colas al que se desea conectar. Managed File Transfer rechaza las conexiones a un servidor SSL incorrecto con un nombre distinguido que no coincida.

Establezca las propiedades **SslTrustStore** y **SslKeyStore** en nombres de archivo que apunten a los archivos de almacén de confianza y de almacén de claves. Si está estableciendo estas propiedades para un agente que ya está en ejecución, detenga y reinicie el agente para reconectarse en modalidad SSL.

Los archivos de propiedades contienen contraseñas de texto sin formato; por consiguiente, contemple a posibilidad de otorgar los permisos correspondientes a los sistemas de archivos.

Si desea más información sobre propiedades SSL, consulte Propiedades SSL para MFT.

5. Si un gestor de colas de agente utiliza SSL, no puede proporcionar los detalles necesarios cuando crea el agente. Para crear el agente, efectúe los pasos siguientes:
   a) Cree el agente utilizando el mandato **fteCreateAgent**. Recibirá un aviso sobre la imposibilidad de publicar la existencia del agente en el gestor de colas de coordinación.
   b) Edite el archivo `agent.properties` que se creó mediante el paso anterior para añadir la información SSL. Cuando el agente se ha iniciado correctamente, se vuelve a intentar la publicación.
6. Si hay agentes o instancias de IBM MQ Explorer en ejecución mientras se modifican las propiedades SSL del archivo `agent.properties` o el archivo `coordination.properties`, debe reiniciar el agente o IBM MQ Explorer.

Referencia relacionada

El archivo MFT `agent.properties`

Conexión a un gestor de colas en modalidad de cliente con autenticación de canal

En IBM WebSphere MQ 7.1 se introdujeron registros de autenticación de canal para controlar con más precisión el acceso a un nivel de canal. Este cambio de comportamiento significa que de forma predeterminada los gestores de colas de IBM WebSphere MQ 7.1 o posteriores recién creados rechazan conexiones de cliente del componente Managed File Transfer.

Si desea más información sobre la autenticación de canal, consulte "Registros de autenticación de canal" en la página 49.

Si la configuración de autenticación de canal del SVRCONN utilizado por Managed File Transfer especifica un ID MCAUSER sin privilegios, hay que otorgar registros de autorización específicos para el gestor de colas, las colas y los temas para que los mandatos y el Managed File Transfer Agent funcionen correctamente. Utilice el mandato de MQSC SET CHLAUTH o el mandato de PCF Establecer registro de autenticación de canal para crear, modificar o eliminar registros de autenticación de canal. Para todos los agentes de Managed File Transfer que desea conectar al gestor de colas IBM WebSphere MQ 7.1 o posterior, puede configurar un ID MCAUSER para utilizarlo para todos los agentes o configurar un ID MCAUSER independiente para cada agente.

Otorgue a cada ID MCAUSER los permisos siguientes:

- Registros de autorización necesarios para el gestor de colas:
  - connect
  - setid
  - inq

- Registros de autorización necesarios para colas.

  Para todas las colas específicas de agente, es decir los nombres de colas que terminan en *agent_name* en la lista siguiente, debe crear estos registros de autorización de cola para cada agente que desea conectar al gestor de colas deIBM WebSphere MQ 7.1 o posterior utilizando una conexión de cliente.

  - put, get, dsp (SYSTEM.DEFAULT.MODEL.QUEUE)
  - put, get, setid, browse (SYSTEM.FTE.COMMAND.*nombre_agente*)
  - put, get (SYSTEM.FTE.DATA.*nombre_agente*)
  - put, get (SYSTEM.FTE.REPLY.*nombre_agente*)
  - put, get, inq, browse (SYSTEM.FTE.STATE.*nombre_agente*)
  - put, get, browse (SYSTEM.FTE.EVENT.*nombre_agente*)
  - put, get (SYSTEM.FTE)
- Registros de autorización necesarios para temas:
  - sub, pub (SYSTEM.FTE)
- Se requieren registros de autorización para las transferencias de archivos.

  Si tiene ID de MCAUSER separados para el agente de origen y de destino, cree los registros de autorización en las colas de los agentes de origen y de destino.

  Por ejemplo, si el ID de MCAUSER del agente de origen es **user1** y el ID de MCAUSER del agente de destino es **user2**, establezca las siguientes autorizaciones para los usuarios del agente:

Usuario AGENT	Cola	Autorización necesaria
user1	SYSTEM.FTE.DATA.*nombre_agente_destino*	put
user1	SYSTEM.FTE.COMMAND.*nombre_agente_destino*	put
user2	SYSTEM.FTE.REPLY.*nombre_agente_origen*	put
user2	SYSTEM.FTE.COMMAND.*nombre_agente_origen*	put

Configuración de SSL o TLS entre el agente de puente Connect:Direct y el nodo Connect:Direct

Configure el agente de puente Connect:Direct y el nodo Connect:Direct para conectarse entre sí a través del protocolo SSL, creando un almacén de claves y un almacén de confianza y estableciendo propiedades en el archivo de propiedades del agente de puente Connect:Direct.

Acerca de esta tarea

Estos pasos incluyen instrucciones para recibir las claves firmadas por una entidad emisora de certificados. Si no utiliza una entidad emisora de certificados, puede generar un certificado autofirmado. Si desea más información sobre cómo generar un certificado firmado automáticamente, consulte "Trabajar con SSL/TLS en UNIX, Linux, and Windows" en la página 289.

Estos pasos incluyen instrucciones para crear un nuevo almacén de claves y un nuevo almacén de confianza para el agente de puente Connect:Direct. Si el agente de puente Connect:Direct ya tiene un almacén de claves y un almacén de confianza que utiliza para conectarse de forma segura a gestores de colas de IBM MQ, puede utilizar el almacén de claves y el almacén de confianza existentes al conectarse de forma segura al nodo Connect:Direct. Para obtener más información, consulte "Configurar el cifrado SSL o TLS para MFT" en la página 562.

Procedimiento

Para el nodo Connect:Direct, realice los pasos siguientes:

1. Genere una clave y un certificado firmado para el nodo Connect:Direct.

Puede hacer esto con la herramienta IBM Key Management que se proporciona con IBM MQ. Para obtener más información, consulte “Trabajar con SSL/TLS” en la página 277.

2. Envíe una solicitud a una entidad emisora de certificados para que le firme la clave. Recibirá a cambio un certificado.
3. Cree un archivo de texto, por ejemplo `/test/ssl/certs/CAcert`, que contenga la clave pública de la entidad emisora de certificados.
4. Instale la Opción Secure+ en el nodo Connect:Direct.

   Si el nodo ya existe, puede instalar la Opción Secure+ ejecutando de nuevo el instalador, especificando la ubicación de la instalación existente y eligiendo instalar sólo la Opción Secure+.
5. Cree un archivo de texto nuevo; por ejemplo, `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt`.
6. Copie el certificado que ha recibido de la autoridad de certificación y la clave privada, que se encuentra en `/test/ssl/cd/privateKeys/`*`node_name`*`.key`, en el archivo de texto.

   El contenido de `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt` debe estar en el formato siguiente:

```
-----BEGIN CERTIFICATE-----
MIICnzCCAgigAwIBAgIBGjANBgkqhkiG9w0BAQUFADBeMQswCQYDVQQGEwJHQjES
MBAGA1UECBMJSGFtcHNoaXJlMRAwDgYDVQQHEwdIdXJzbGV5MQwwCgYDVQQKEwNJ
Qk0xDjAMBgNVBAsTBU1RSVBUMQswCQYDVQQDEwJDQTAeFw0xMTAzMDExNjIwNDZa
Fw0yMTAyMjYxNjIwNDZaMFAxCzAJBgNVBAYTAkdCMRIwEAYDVQQIEwlIYW1wc2hp
cmUxDDAKBgNVBAoTA0lCTTEOMAwGA1UECxMFTVFGVEUxDzANBgNVBAMTBmJpbmJh
ZzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAvgP1QIklU9ypSKD1XoODo1yk
EyMFXBOUpZRrDVxjoSEC0vtWNcJ199e+Vc4UpNybDyBu+NkDlMNofX4QxeQcLAFj
WnhakqCiQ+JIAD5AurhnrwChe0MV3kjA84GKH/rOSVqtl984mu/lDyS819XcfSSn
cOOMsK1KbneVSCIV2XECAwEAAaN7MHkwCQYDVR0TBAIwADAsBglghkgBhvhCAQ0E
HxYdT3BlblNTTCBHZW5lcmF0ZWQgQ2VydGlmaWNhdGUwHQYDVR0OBBYEFNXMIpSc
csBXUniW4A3UrZnCRsv3MB8GA1UdIwQYMBaAFDXY8rmj4lVz5+FVAoQb++cns+B4
MA0GCSqGSIb3DQEBBQUAA4GBAFc7klXa4pGKYgwchxKpE3ZF6FNwy4vBXS216/ja
8h/vl8+iv01OCL8t0ZOKSU95fyZLzOPKnCH7v+ItFSE3CIiEk9Dlz2U6WO9lICwn
l7PL72TdfaL3kabwHYVf17IVcuL+VZsZ3HjLggP2qHO9ZuJPspeT9+AxFVMLiaAb
8eHw
-----END CERTIFICATE-----
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,64A02DA15B6B6EF9

57kqxLOJ/gRUOIQ6hVK2YN13B4E1jAi1gSme0I5ZpEIG8CHXISKB7/0cke2FTqsV
lvI99QyCxsDWoMNt5fj51v7aPmVeS60bOm+UlGre8B/Ze18JVj2O4K2Uh72rDCXE
5e6eFxSdUM207sQDy20euBVELJtM2kOkL1ROdoQQSlU3XQNgJw/t3ZIx5hPXWEQT
rjRQO64BEhb+PzzxPF8uwzZ9IrUK9BJ/UUnqC6OdBR87IeA4pnJD1Jvb2ML7EN9Z
5Y+50hTKI8OGvBvWXO4fHyvIX5aslwhBoArXIS1AtNTrptPvoaP1zyIAeZ6OCVo/
SFo+A2UhmtEJeOJaZG2XZ3H495fAw/EHmjehzIACwukQ9nSIETgu4A1+CV64RJED
aYBCM8UjaAkbZDH5gn7+eBov0ssXAXWDyJBVhUOjXjvAj/e1h+kcSF1hax5D//AI
66nRMZzboSxNqkjcVd8wfDwP+bEjDzUaaarJTS7lIFeLLw7eJ8MNAkMGicDkycL0
EPBU9X5QnHKLKOfYHN/1WgUk8qt3UytFXXfzTXGF3EbsWbBupkT5e5+lYcX8OVZ6
sHFPNlHluCNy/riUcBy9iviVeodX8IomOchSyO5DKl8bwZNjYtUP+CtYHNFU5BaD
I+1uUOAeJ+wjQYKT1WaeIGZ3VxuNITJul8y5qDTXXfX7vxM5OoWXa6U5+AYuGUMg
/itPZmUmNrHjTk7ghT6i1IQOaBowXXKJBlMmq/6BQXN2IhkD9ys2qrvM1hdi5nAf
egmdiG50loLnBRqWbfR+DykpAhK4SaDi2F52Uxovw3Lhiw8dQP7lzQ==
-----END RSA PRIVATE KEY-----
```

7. Inicie la Herramienta de administración Secure+.
   - En sistemas Linux o UNIX, ejecute el mandato **spadmin.sh**.
   - En sistemas Windows, pulse **Inicio** > **Programas** > **Sterling Commerce Connect: Direct** > **Herramienta de administración CD Secure+**

   Se inicia la Herramienta de administración CD Secure+.
8. En la Herramienta de administración CD Secure+, efectúe una doble pulsación en la línea **.Local** para editar la configuración SSL o TLS principal.
   a) Seleccione **Habilitar protocolo SSL** o **Habilitar protocolo TLS**, dependiendo del protocolo que esté utilizando.
   b) Seleccione **Inhabilitar alteración temporal**.
   c) Seleccione al menos una suite de cifrado.
   d) Si desea autenticación bidireccional, cambie el valor de **Habilitar autenticación de cliente** a Yes.

e) En el campo **Certificado raíz de confianza**, especifique la vía de acceso al archivo de certificado público de la entidad emisora de certificados, `/test/ssl/certs/CAcert`.

f) En el campo **Archivo de certificado de clave** , especifique la vía de acceso al archivo que ha creado, `/test/ssl/cd/keyCertFile/`*`node_name`*`.txt`.

9. Efectúe una doble pulsación en la línea **.Client** para editar la configuración SSL o TLS principal.

a) Seleccione **Habilitar protocolo SSL** o **Habilitar protocolo TLS**, dependiendo del protocolo que esté utilizando.

b) Seleccione **Inhabilitar alteración temporal**.

Para el agente de puente Connect:Direct, realice los pasos siguientes:

10. Cree un almacén de confianza. Puede hacer esto creando una clave ficticia y suprimiendo luego la clave ficticia.

Puede utilizar los siguientes mandatos:

```
keytool -genkey -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

```
keytool -delete -alias dummy -keystore /test/ssl/fte/stores/truststore.jks
```

11. Importe el certificado público de la entidad emisora de certificados al almacén de confianza.

Puede utilizar el siguiente mandato:

```
keytool -import -trustcacerts -alias myCA
        -file /test/ssl/certs/CAcert
        -keystore /test/ssl/fte/stores/truststore.jks
```

12. Edite el archivo de propiedades del agente de puente Connect:Direct.

Incluya las siguientes líneas en cualquier parte del archivo:

```
cdNodeProtocol=protocol
cdNodeTruststore=/test/ssl/fte/stores/truststore.jks
cdNodeTruststorePassword=password
```

En el ejemplo de este paso, *protocolo* es el protocolo que está utilizando, ya sea SSL o TLS, y *contraseña* es la contraseña que especificó cuando creó el almacén de confianza.

13. Si desea autenticación bidireccional, cree una clave y un certificado para el agente de puente Connect:Direct.

a) Cree un almacén de claves y una clave.

Puede utilizar el siguiente mandato:

```
keytool -genkey -keyalg RSA -alias agent_name
        -keystore /test/ssl/fte/stores/keystore.jks
        -storepass password -validity 365
```

b) Genere una solicitud de firma.

Puede utilizar el siguiente mandato:

```
keytool -certreq -v -alias agent_name
        -keystore /test/ssl/fte/stores/keystore.jks -storepass password
        -file /test/ssl/fte/requests/agent_name.request
```

c) Importe el certificado que reciba del paso anterior al almacén de claves. El certificado debe estar en formato x.509.

Puede utilizar el siguiente mandato:

```
keytool -import -keystore /test/ssl/fte/stores/keystore.jks
        -storepass password -file certificate_file_path
```

d) Edite el archivo de propiedades del agente de puente Connect:Direct.

Incluya las siguientes líneas en cualquier parte del archivo:

```
cdNodeKeystore=/test/ssl/fte/stores/keystore.jks
cdNodeKeystorePassword=password
```

En el ejemplo de este paso, *contraseña* es la contraseña que especificó cuando creó el almacén de claves.

Tareas relacionadas

Configurar el puente Connect:Direct

ULW Protección de clientes de AMQP

Puede utilizar una serie de mecanismos de seguridad para proteger las conexiones de los clientes AMQP y asegurarse de que los datos están protegidos adecuadamente en la red. Puede crear seguridad en las aplicaciones de MQ Light. También puede utilizar las funciones de seguridad existentes de IBM MQ con clientes AMQP, de la misma forma que se utilizan las características para otras aplicaciones.

Reglas de autenticación de canal (CHLAUTH)

Puede utilizar las reglas de autenticación de canal para restringir las conexiones TCP al gestor de colas. Los canales AMQP admiten el uso de reglas de autenticación de canal que puede configurar para el gestor de colas. Si las reglas de autenticación de canal se definen con un perfil que coincide con los canales AMQP en el gestor de colas, estas reglas se aplican a los canales. De forma predeterminada, la autenticación de canal está habilitada en nuevos gestores de colas de IBM® MQ por lo que debe completar al menos alguna configuración antes de poder utilizar el canal AMQP.

Para obtener más información sobre cómo configura reglas de autenticación de canal para permitir conexiones AMQP al gestor de colas, consulte Creación y utilización de canales AMQP.

Autenticación de conexión (CONNAUTH)

Puede utilizar la autenticación de conexión para autenticar las conexiones a un gestor de colas. Los canales AMQP admiten el uso de la autenticación de conexión para controlar el acceso al gestor de colas de las aplicaciones AMQP.

El protocolo AMQP utiliza la infraestructura SASL (capa de seguridad y autenticación simple) para especificar cómo se autentica una conexión. Hay varios mecanismos SASL y IBM MQ admite dos mecanismos SASL: ANONYMOUS y PLAIN.

En el caso de ANONYMOUS, no se pasa ninguna credencial del cliente al gestor de colas para la autenticación. Si el objeto MQ AUTHINFO especificado en el atributo CONNAUTH tiene un valor de CHCKCLNT de REQUIRED o REQDADM (si se conecta como un usuario administrativo), se rechaza la conexión. Si el valor de CHCKCLNT es NONE u OPTIONAL, se acepta la conexión.

En el caso de PLAIN, se pasa el nombre de usuario y la contraseña del cliente al gestor de colas para la autenticación. Si el objeto MQ AUTHINFO especificado en el atributo CONNAUTH tiene un valor de CHCKCLNT de NONE, se rechaza la conexión. Si el valor de CHCKCLNT es OPTIONAL, REQUIRED o REQDADM (si se conecta como un usuario administrativo), el gestor de colas comprueba el nombre de usuario y la contraseña. El gestor de colas comprueba el sistema operativo (si el objeto AUTHINFO es de tipo IDPWOS) o un repositorio LDAP (si el objeto AUTHINFO es de tipo IDPWLDAP).

En la tabla siguiente se resume este comportamiento de autenticación:

Tabla 95. Resumen de los mecanismos SASL y la autenticación de conexión		
Mecanismo SASL	**¿Se pasan las credenciales del cliente al gestor de colas?**	**Valor de CHKCLNT**
ANONYMOUS	No	REQUIRED o REQDADM: se rechaza la conexión NONE o OPTIONAL: se acepta la conexión
PLAIN	Sí, nombre de usuario y la contraseña	REQUIRED REQDADM u OPTIONAL: el gestor de colas comprueba el nombre de usuario y la contraseña NONE: se rechaza la conexión

Si utiliza un cliente de MQ Light, puede especificar las credenciales incluyéndolas en la dirección de AMQP a la que se conecta, por ejemplo:

```
amqp://mwhitehead:mYp4ssw0rd@localhost:5672/sports/football
```

Valor de MCAUSER en un canal

Los canales AMQP tienen un atributo MCAUSER, que puede utilizar para establecer el ID de usuario de IBM MQ bajo el que todas las conexiones a dicho canal tienen autorización. Todas las conexiones de clientes AMQP a dicho canal adoptan el ID de MCAUSER configurado. Ese ID de usuario se utiliza para la autorización de mensajería en diferentes temas.

Se recomienda utilizar la autenticación de canal (CHLAUTH) para proteger las conexiones a gestores de colas. Si está utilizando la autenticación de canal, se recomienda configurar el valor de MCAUSER para un usuario sin privilegios. Así se garantiza que si una conexión a un canal no coincide con una regla CHLAUTH, no se autoriza la conexión a realizar operaciones de mensajería en el gestor de colas.

Nota: Windows En Windows, antes de IBM MQ 9.1.1, el valor del ID de usuario MCAUSER solo está soportado para los ID de usuario de hasta 12 caracteres de longitud.

V 9.1.1 Desde IBM MQ 9.1.1, se elimina el límite de 12 caracteres.

soporte SSL/TLS

Los canales AMQP admiten el cifrado SSL/TLS utilizando claves del repositorio de claves configurado para el gestor de colas. Las opciones de configuración de canal AMQP para el cifrado SSL/TLS admiten las mismas opciones que otros tipos de canal MQ; puede indicar una especificación de cifrado y si el gestor de colas requiere certificados de las conexiones de cliente AMQP.

Mediante el uso de atributos FIPS del gestor de colas puede controlar las suites de cifrado SSL/TLS, que puede utilizar para proteger las conexiones de clientes AMQP.

Para obtener información sobre cómo configurar un repositorio de claves para el gestor de colas, consulte Trabajar con SSL o TLS en sistemas UNIX, Linux y Windows.

Para obtener información sobre cómo configurar el soporte SSL/TLS para una conexión de cliente AMQP, consulte Creación y utilización de canales AMQP.

Servicio de autenticación y autorización Java (JAAS)

Puede configurar opcionalmente canales AMQP con un módulo de inicio de sesión JAAS, que puede comprobar el nombre de usuario y la contraseña proporcionados por un cliente AMQP. Consulte "Configuración de JAAS para canales AMQP" en la página 570.

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Creación y utilización de canales AMQP

ULW Restricción de la toma de control del cliente AMQP

Cuando se realiza una conexión de cliente AMQP que tiene el mismo identificador de cliente que una conexión de cliente AMQP existente, la conexión de cliente existente se desconecta de forma predeterminada. Sin embargo, puede configurar el gestor de colas para restringir el comportamiento de toma de control del cliente para que la toma de control sea posible solo cuando se cumplan determinados criterios.

Por ejemplo, es posible que no sea adecuado desconectar la conexión de cliente existente si hay distintos equipos que están desarrollando aplicaciones AMQP y puede que estén utilizando el mismo ID de cliente. Para resolver este problema puede restringir la toma de control del cliente basándose en el nombre del canal AMQP que se utiliza, la dirección IP del cliente y el ID de usuario del cliente (cuando está habilitada la autenticación SASL).

Utilice los valores de los atributos de gestor de colas **AdoptNewMCA** y **AdoptNewMCACheck** para especificar el nivel necesario de restricción de toma de control, tal como se detalla en la tabla siguiente:

*Tabla 96. Valores de **AdoptNewMCA** y **AdoptNewMCACheck** para restringir la toma de control del cliente*

AdoptNewMCA	**AdoptNewMCACheck**	**Comprobación de criterios antes de que se permita la toma de control del cliente**
NO o sin definir	No aplicable	Ninguno. La toma de control del cliente está permitida para todas las conexiones de cliente que están autenticadas y pasan todas las reglas de CHLAUTH.
ALL (o un valor distinto de NO)	QM o sin definir	Ninguno. La toma de control del cliente está permitida para todas las conexiones de cliente que están autenticadas y pasan todas las reglas de CHLAUTH.
ALL (o un valor distinto de NO)	NOMBRE	ID de usuario (cuando SASL está habilitado) Nombre de canal
ALL (o un valor distinto de NO)	ADDRESS	ID de usuario (cuando SASL está habilitado) Dirección IP
ALL (o un valor distinto de NO)	ALL	ID de usuario (cuando SASL está habilitado) Nombre de canal Dirección IP

Los atributos del gestor de colas **AdoptNewMCA** y **AdoptNewMCACheck** forman parte de la configuración del gestor de colas, que se define en la stanza CHANNELS. En IBM MQ para Windows y IBM MQ para sistemas Linux x86-64, modifique la información de configuración utilizando IBM MQ Explorer. En otros sistemas, modifique la información editando el archivo de configuración `qm.ini`. Para obtener información sobre cómo modificar la información de los canales del gestor de colas, consulte Atributos de canales.

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Creación y utilización de canales AMQP

ULW Configuración de JAAS para canales AMQP

Los módulos personalizados JAAS (Java Authentication and Authorization Service) se pueden utilizar para autenticar credenciales de nombre de usuario y contraseña pasadas a un canal AMQP por un cliente AMQP cuando se conecta.

Acerca de esta tarea

Es posible que desee utilizar un módulo JAAS personalizado si ya utiliza módulos JAAS para la autenticación en otros sistemas basados en Java y desea reutilizar estos módulos para autenticar las conexiones AMQP con MQ. De forma alternativa, es posible que desee escribir un módulo JAAS personalizado si las características de autenticación creadas en MQ no dan soporte al mecanismo de autenticación que desea utilizar.

La configuración de los módulos JAAS para los canales AMQP se realiza a un nivel de gestor de colas. Esto significa que, si configura un módulo JAAS para autenticar las conexiones AMQP con el gestor de colas, el módulo se aplicará a todos los canales AMQP. El nombre del canal que ha invocado el módulo JAAS se pasa al módulo, lo que le permite codificar un registro de JAAS diferente en el comportamiento para diferentes canales.

Otra información también se pasa al módulo JAAS:

- El ID de cliente del cliente AMQP que está intentando autenticarse.
- La dirección de red del cliente AMQP.
- El nombre del canal que ha invocado el módulo JAAS.

Procedimiento

Puede configurar un módulo de configuración JAAS para los canales AMQP completando los pasos siguientes:

1. Defina un archivo `jaas.config` que contenga una o más stanzas de configuración de módulo JAAS. La stanza debe especificar el nombre completo de la clase Java que implementa la interfaz `javax.security.auth.spi.LoginModule` de JAAS.
   - Un archivo `jaas.config` predeterminado se suministra con el producto y se encuentra en *QM_data_directory*`/amqp/jaas.config`.
   - Ya se ha definido una stanza preconfigurada denominada `MQXRConfig` en el archivo `jaas.config` predeterminado.
2. Especifique el nombre de la stanza que se va a utilizar para los canales AMQP.
   - UNIX Añada una propiedad al archivo `amqp_unix.properties`.
   - Windows Añada una propiedad al archivo `amqp_win.properties`.

   La propiedad tiene el formato siguiente:

```
com.ibm.mq.MQXR.JAASConfig=JAAS_stanza_name
```

Por ejemplo:

```
com.ibm.mq.MQXR.JAASConfig=MQXRConfig
```

3. Configure el entorno del gestor de colas para que incluya la clase del módulo personalizado. El servicio AMQP debe tener acceso a la clase Java configurada en la stanza de configuración JAAS.

   Para ello, añada la vía de acceso a la clase JAAS al archivo `service.env` de MQ. Edite el archivo `service.env` en el directorio de configuración de MQ (*directorio_config_MQ*) o el directorio de configuración del gestor de colas *directorio_config_MQ*) para establecer la variable CLASSPATH en la ubicación de la clase de módulo JAAS.

Qué hacer a continuación

Se proporciona un módulo de inicio de sesión JAAS de ejemplo con el producto en el directorio *mq_installation_directory*`/amqp/samples` . El módulo de inicio de sesión JAAS de ejemplo autentica todas las conexiones cliente, independientemente del nombre de usuario o de la contraseña con el que se conecta el cliente.

Puede modificar el código fuente del ejemplo y volver a compilarlo para intentar autenticar únicamente a los usuarios específicos con una contraseña determinada. Para configurar el canal AMQP en un sistema UNIX para que utilice el módulo de inicio de sesión JAAS de ejemplo suministrado con el producto:

1. Edite el archivo `/var/mqm/qmgrs/QMNAME/amqp/amqp_unix.properties` y establezca la propiedad `com.ibm.mq.MQXR.JAASConfig=MQXRConfig`.
2. Edite el archivo /var/mqm/service.env y establezca la propiedad `CLASSPATH=`*mq_installation_location*`/amqp/samples`

El archivo `jaas.config` ya contiene una stanza denominada `MQXRConfig` que especifica la clase de ejemplo `samples.JAASLoginModule` como clase de módulo de inicio de sesión. No es necesario ningún cambio para `jaas.config` antes de probar el módulo de ejemplo.

Tareas relacionadas

Desarrollo de aplicaciones cliente AMQP

Creación y utilización de canales AMQP

Advanced Message Security

Advanced Message Security (AMS) es un componente de IBM MQ que proporciona un alto nivel de protección para los datos confidenciales que fluyen a través de la red IBM MQ, aunque no afecta a las aplicaciones finales.

Visión general de Advanced Message Security

Las aplicaciones de IBM MQ pueden utilizar Advanced Message Security para enviar datos sensibles, tales como transacciones financieras de alto valor e información personal, con niveles diferentes de protección utilizando un modelo de criptografía de clave pública.

Referencia relacionada

Códigos de retorno de GSKit utilizados en mensajes de AMS

Funciones y características de Advanced Message Security

Advanced Message Security amplía los servicios de seguridad de IBM MQ para proporcionar funciones de firma y cifrado de los datos a nivel de mensaje. Los servicios ampliados garantizan que los datos de los mensajes no se han modificado entre el momento en que se colocaron originalmente en una cola y cuando se recuperaron. Además, AMS verifica que el emisor de los datos de un mensaje está autorizado para colocar mensajes firmados en una cola de destino.

AMS proporciona las siguientes funciones:

- Protege las transacciones sensibles o de alto valor procesadas por IBM MQ.

- Detecta y elimina mensajes no autorizados antes de que sean procesados por una aplicación receptora.
- Verifica que los mensajes no se han modificado mientras estaban en tránsito entre una cola y otra.
- Protege los datos no sólo mientras circulan por la red, sino también cuando se colocan en una cola.
- Protege las aplicaciones propietarias y escritas por el cliente existentes para IBM MQ.
- z/OS V 9.1.3 A partir de IBM MQ 9.1.3, IBM MQ for z/OS proporciona la posibilidad de eliminar y añadir, de forma opcional, la protección de AMS en los mensajes que fluyen a través de la red, respectivamente. Esto se conoce como *Intercepción del agente de canal de mensajes (MCA) de servidor a servidor*..
- ULW V 9.1.0.4 V 9.1.4 A partir de IBM MQ 9.1.4 y IBM MQ 9.1.0 Fix Pack 4, se ha añadido una comprobación al código de la biblioteca de IBM MQ que se ejecuta en el programa de aplicación del cliente. La comprobación se ejecuta pronto en su inicialización para leer el valor de la variable de entorno *AMQ_AMS_FIPS_OFF* y, si se establece en cualquier valor, el código de GSKit se ejecutará en modalidad no FIPS en esa aplicación.

Calidades de protección disponibles con AMS

Existen tres calidades de protección para Advanced Message Security, `Integrity`, `Privacyy` `Confidentiality`.

La protección `Integrity` se ofrece mediante la firma digital, que proporciona una garantía sobre quién ha creado el mensaje y que dicho mensaje no se ha modificado ni se ha manipulado indebidamente.

La protección `Privacy` se proporciona mediante una combinación de firma digital y cifrado. El cifrado asegura que los datos del mensaje solo pueda visualizarlos el destinatario o destinatarios deseados. Aunque existan destinatarios no autorizados que obtengan una copia de los datos del mensaje cifrados, no podrán ver los datos reales del mensaje.

La protección de `Confidentiality` se proporciona mediante el cifrado sólo con la reutilización de claves opcional.

Efecto sobre el rendimiento

AMS utiliza una combinación de rutinas criptográficas simétricas y asimétricas para proporcionar la firma digital y el cifrado. Debido a que las operaciones de clave simétrica son muy rápidas en comparación con las operaciones de clave asimétrica, que son intensivas para la CPU, esto puede tener un impacto significativo en los costes de protección de un gran número de mensajes con AMS.

Rutinas criptográficas asimétricas

Por ejemplo, al transferir un mensaje firmado, el código hash del mensaje se firma utilizando una operación de clave asimétrica.

Al obtener un mensaje firmado, se utiliza una operación de clave asimétrica más para verificar el código hash firmado.

Por lo tanto, se necesita un mínimo de dos operaciones de clave asimétrica por mensaje para firmar y verificar los datos del mensaje.

Rutinas criptográficas asimétricas y simétricas

Al transferir un mensaje cifrado, se genera una clave simétrica y luego se cifra utilizando una operación de clave asimétrica para cada destinatario deseado del mensaje.

Los datos del mensaje se cifran a continuación con la clave simétrica. Al obtener el mensaje cifrado, los destinatarios deseados deben utilizar una operación de clave asimétrica para descubrir la clave simétrica que se utiliza para el mensaje.

Por lo tanto, las tres calidades de protección contienen diversos elementos de las operaciones de clave asimétricas que ocupan muchos recursos de CPU, lo cual afectará de forma significativa a la tasa máxima de transferencia de mensajes que se puede obtener en las aplicaciones que transfieren y obtienen mensajes.

Las políticas de `Confidentiality` permiten, no obstante, la reutilización de la clave simétrica en una secuencia de mensajes. Se pueden realizar ahorros de costes de CPU significativos con las políticas de `Confidentiality` a través de la reutilización de claves simétricas. Esta modalidad de operación continúa utilizando el formato PKCS#7 para compartir una clave de cifrado simétrica. No obstante, no hay ninguna firma digital, lo que elimina algunas de las operaciones por clave asimétrica de mensaje. La clave simétrica se debe continuar cifrando con las operaciones de claves asimétricas para cada destinatario, pero la clave simétrica se puede reutilizar opcionalmente en varios mensajes destinados a los mismos destinatarios. Si la política permite la reutilización de claves, solo el primer mensaje requiere operaciones de claves asimétricas. Los mensajes siguientes solo necesitan utilizar operaciones de claves simétricas.

Reutilización de claves

Con las políticas de `Confidentiality` , puede utilizar el enfoque de reutilización de claves simétricas para reducir significativamente los costes implicados en el cifrado de un número de mensajes que se colocan en la misma cola y están pensados para el mismo destinatario o destinatarios.

Por ejemplo, al transferir 10 mensajes cifrados al mismo conjunto de destinatarios, se genera una clave simétrica y, a continuación, se cifra para el primer mensaje, utilizando una operación de clave asimétrica para cada destinatario del mensaje.

En función de los límites controlados por la política, la clave simétrica cifrada se puede reutilizar en los mensajes posteriores que estén dirigidos a los mismos destinatarios. Una aplicación que obtiene mensajes cifrados puede aplicar la misma optimización, con lo cual la aplicación puede detectar cuándo no se ha modificado una clave simétrica y evitar el gasto de recuperar la clave simétrica.

En este ejemplo, el 90% de las operaciones de clave asimétrica se pueden evitar en las aplicaciones de transferencia y de obtención mediante la reutilización de la misma clave.

Para obtener más información sobre cómo realizar la reutilización de clave, consulte:

- Mandato MQSC SET POLICY
- Mandato de control setmqspl
- IBM i Mandato de IBM i SETMQMSPL

Conceptos esenciales de AMS

Conozca los conceptos esenciales de Advanced Message Security para comprender cómo trabaja la herramienta y cómo utilizarla de forma efectiva.

Infraestructura de claves públicas y Advanced Message Security

Una infraestructura de claves públicas (PKI) es un sistema de recursos, políticas y servicios que permiten utilizar la criptografía de clave pública para lograr una comunicación segura.

No existe un estándar individual que defina los componentes de una infraestructura de claves públicas (PKI), pero normalmente una PKI utiliza certificados de clave pública y comprende entidades emisoras de certificados (CA) y otras entidades de registro (RA) que proporcionan los servicios siguientes:

- Emisión de certificados digitales
- Validación de certificados digitales
- Revocación de certificados digitales
- Distribución de certificados

La identidad de los usuarios y las aplicaciones está representada por el campo **nombre distinguido (DN)** en un certificado asociado con mensajes firmados o cifrados. Advanced Message Security utiliza esta identidad para representar un usuario o una aplicación. Para autenticar esta identidad, el usuario o aplicación debe tener acceso al almacén de claves donde se almacenan el certificado y la clave privada asociada. Cada certificado está representado por una etiqueta en el almacén de claves.

Conceptos relacionados

“Utilización de almacenes de claves y certificados” en la página 616

Para proporcionar protección de cifrado transparente para las aplicaciones de IBM MQ, Advanced Message Security utiliza el archivo de almacén de claves, donde se almacenan certificados de clave pública y una clave privada. En z/OS, se utiliza un conjunto de claves SAF en lugar del archivo del almacén de claves.

Certificados digitales en AMS

Advanced Message Security asocia usuarios y aplicaciones con certificados digitales X.509 estándar. Normalmente los certificados X.509 están firmados por una entidad emisora de certificados fiable y supone la utilización de claves privadas y públicas para el cifrado y descifrado.

Los certificados digitales proporcionan protección frente a la suplantación de identidad mediante la asociación de una clave pública con su propietario, ya sea un individuo, un gestor de colas o alguna otra entidad. Los certificados digitales también se conocen como certificados de clave pública, pues garantizan la propiedad de una clave pública cuando se utiliza un sistema de claves asimétricas. Este sistema requiere crear clave pública y una clave privada para una aplicación. Los datos cifrados mediante la clave pública sólo se pueden descifrar mediante la clave privada correspondiente, mientras que los datos cifrados mediante la clave privada sólo se pueden descifrar mediante la clave pública correspondiente. La clave privada se almacena en un archivo de base de datos de claves protegido por contraseña. Sólo el propietario tiene acceso a la clave privada que se utiliza para descifrar los mensajes cifrados mediante la clave pública correspondiente.

Si las claves públicas las envía directamente su propietario a otra entidad, existe el riesgo de que el mensaje pueda ser interceptado y de que la clave pública sea sustituida por otra. Esto se conoce como ataque de interceptor. La solución es intercambiar claves públicas a través de un agente fiable, con lo que el usuario tiene una mayor garantía de que la clave pública pertenece a la entidad con la que se está comunicando. En lugar de enviar la clave pública directamente, el usuario solicita a un agente fiable que la incorpore a un certificado digital. El agente fiable que emite certificados digitales se denomina entidad emisora de certificados.

Para obtener más información sobre los certificados digitales, consulte ¿Qué es un certificado digital?

Un certificado digital contiene la clave pública de una entidad y declara que la clave pública pertenece a esa entidad:

- cuando un certificado es para una entidad individual, se denomina *certificado personal* o *certificado de usuario*.
- cuando un certificado es para una entidad emisora de certificados, el certificado se denomina *certificado de CA* o *certificado de firmante*.

Nota: Advanced Message Security da soporte a los certificados autofirmados en las aplicaciones Java y nativas

Conceptos relacionados

“Criptografía” en la página 7
El cifrado es el proceso de convertir texto legible, denominado *texto plano*, en un formato ilegible, denominado *texto cifrado*.

Multi *Gestor de autorizaciones sobre objetos*

En Multiplatforms, el gestor de autorizaciones sobre objetos (OAM) es el componente de servicio de autorización que se suministra con los productos IBM MQ.

El acceso a las entidades de Advanced Message Security se controla mediante grupos de usuarios de IBM MQ y el OAM. Los administradores pueden utilizar la interfaz de línea de mandatos para otorgar o revocar autorizaciones según sea necesario. Grupos de usuarios diferentes pueden tener clases diferentes de autorización de acceso para unos mismos objetos. Por ejemplo, un grupo puede realizar operaciones PUT y GET para una cola determinada, mientras que otro grupo puede tener permiso sólo para examinar la cola. De la misma manera, algunos grupos pueden tener autorización GET y PUT para una cola, pero no pueden modificar ni suprimir la cola.

Mediante el OAM, puede controlar lo siguiente:

- Acceso a objetos de Advanced Message Security a través de la interfaz de cola de mensajes (MQI). Cuando un programa de aplicación intenta acceder a objetos, el OAM comprueba si el perfil de usuario que realiza la solicitud tiene autorización para la operación solicitada. Esto significa que las colas y los mensajes de las colas se pueden proteger contra el acceso no autorizado.
- El permiso para utilizar mandatos PCF y MQSC.

Conceptos relacionados

Gestor de autorizaciones sobre objetos

Descripción general de la interfaz de cola de mensajes (Message Queue Interface, MQI)

Tecnología soportada por Advanced Message Security

Advanced Message Security depende de varios componentes de tecnología para proporcionar una infraestructura de seguridad.

Advanced Message Security es compatible con las siguientes interfaces de programación de aplicaciones (API) de IBM MQ:

- Interfaz de cola de mensajes (MQI)
- IBM MQ Java Message Service (JMS) 1.0.2 y 1.1.
- Clases base de IBM MQ para Java
- Clases de IBM MQ para .Net en modalidad no gestionada

Nota: Advanced Message Security es compatible con las entidades emisoras de certificados que cumplen la especificación X.509.

Limitaciones conocidas de AMS

Existe un número de opciones de IBM MQ que no están soportadas o que tienen limitaciones para Advanced Message Security.

- Las siguientes opciones de IBM MQ no están soportadas o tienen limitaciones:

  Publicación/suscripción
  Una de las principales ventajas de un modelo de mensajería publicación/suscripción respecto de un modelo punto a punto es que las aplicaciones emisora y receptora no necesitan saber nada la una de la otra para que los datos que se envíen y reciban. Esta ventaja se pierde con el uso de políticas Advanced Message Security en las que hay que definir los destinatarios o firmantes autorizados. Es posible que una aplicación publique en un tema a través de una definición de cola de alias que esté protegida por una política, también es posible que una aplicación suscriptora obtenga mensajes de una cola protegida por política. No es posible asignar una política directamente a una cadena de tema, las políticas solo pueden asignarse a definiciones de cola.

  Conversión de datos de canal
  La carga útil protegida de un mensaje protegido de Advanced Message Security se transmite en formato binario, lo que garantiza que la conversión de datos en un canal entre aplicaciones no invalide el resumen de mensaje (digest). Las aplicaciones que recuperan mensajes de una cola protegida por política tienen que solicitar una conversión de datos, la conversión de la carga útil protegida se intentará una vez verificados y desprotegidos correctamente los mensajes.

  Listas de distribución
  Las políticas de Advanced Message Security pueden usarse al proteger aplicaciones colocando mensajes en listas de distribución, siempre que cada cola de destino de la lista tenga definida una política idéntica. Si se identifican políticas incoherentes cuando una aplicación abre una lista de distribución, la operación de apertura (open) fallará y se devolverá un error de seguridad a la aplicación.

  Segmentación de mensajes de aplicación
  El tamaño de los mensajes protegidos por política aumentará y no es posible que las aplicaciones especifiquen con precisión los límites de segmento de un mensaje.

Aplicaciones que utilizan IBM MQ classes for .NET en un nodo gestionado (conexiones de cliente)
Las aplicaciones que usan IBM MQ classes for .NET en modo gestionado (conexiones de cliente) no están soportadas.

Nota: La intercepción MCA se puede utilizar para permitir a clientes no soportados utilizar AMS.

El cliente de servicio de mensajería para aplicaciones .NET (XMS) en una modalidad gestionada
Los clientes de servicio de mensajería de aplicaciones .NET (XMS) en modo gestionado no están soportados.

Nota: La intercepción MCA puede utilizarse para permitir que los clientes soportados usen AMS.

Colas IBM MQ procesadas por el puente IMS
Las colas IBM MQ procesadas por el puente IMS no están soportadas.

Nota: AMS está soportado en colas Puente CICS. Debe utilizar el mismo ID de usuario para MQPUT (cifrar) y MQGET (descifrar) en colas Puente CICS.

Colocación en espera de método de obtención
La colocación en espera del método de obtención no está soportada para las aplicaciones de obtención respecto a colas que tienen políticas AMS definidas para ellas.

V 9.1.3 **Intercepción de agente de canal de mensajes de servidor a servidor**
A partir de IBM MQ 9.1.3, en IBM MQ for z/OS, la intercepción de agente de canal de mensajes de servidor a servidor solo se admite para los tipos de canal emisor, servidor, receptor y solicitante.

- Los usuarios deben evitar colocar más de un certificado con el mismo nombre distinguido en un único archivo de almacén de claves, porque la selección de qué certificado usar al proteger un mensaje no está definida.
- AMS no está soportado en JMS si la propiedad **WMQ_PROVIDER_VERSION** se establece en 6.
- El interceptor AMS no está soportado para canales AMQP o MQTT.

z/OS V 9.1.3 Visión general de la intercepción de Advanced Message Security en los canales de mensajes

En z/OS, la interceptación de Advanced Message Security (AMS) mejora la oferta existente añadiendo una opción adicional de protección de política de seguridad (SPLPROT) a los canales emisor, servidor, receptor y peticionario.

Actualmente, utilizando el ejemplo de las comunicaciones de un centro de intercambio de información con un banco, ambos extremos del sistema deben dar soporte a AMS, tal como se muestra en la Figura 1.

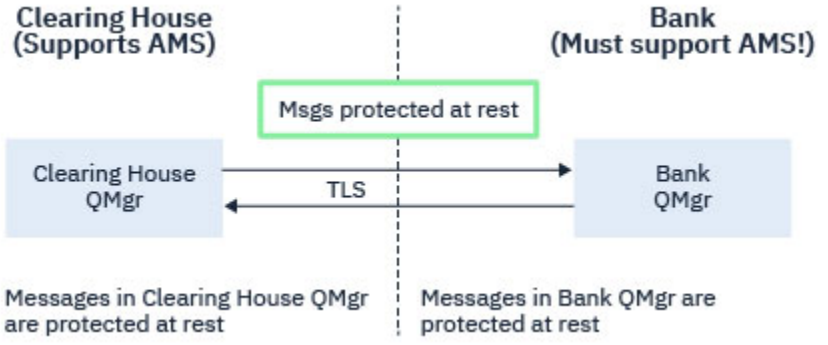


Figura 32. Uso actual de AMS

Una ventaja clave de la opción adicional es que si la empresa tiene AMS configurado, y no todos los business partners dan soporte a AMS, puede eliminar la protección de los mensajes de salida y proteger los mensajes de entrada en los canales hacia y desde los business partners que no dan soporte a AMS.

Utilizando el ejemplo de un centro de intercambio de información y los bancos, este escenario se muestra en la Figura 2, donde hay un flujo de mensajes entre el centro de intercambio de información, los bancos y los business partners donde algunas instituciones tienen AMS y otras no.

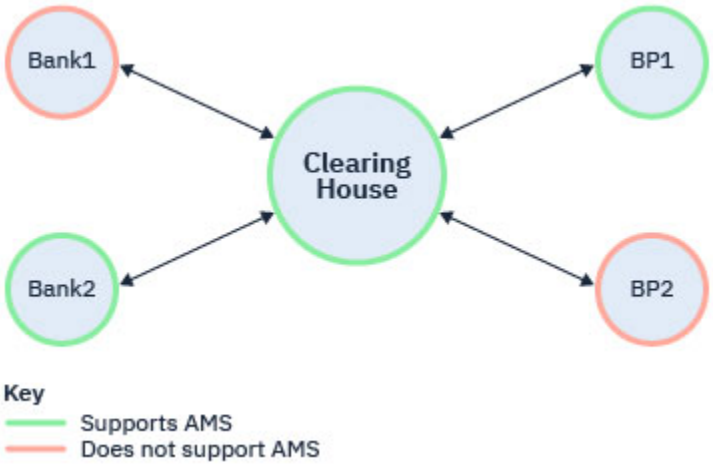


Figura 33. Algunos partners dan soporte a AMS y otros no

Normalmente, los canales están habilitados para TLS.

Sin embargo, podría darse el caso de que algunos bancos y business partners no den soporte a AMS y sea necesario poder intercambiar mensajes entre todos los bancos y business partners. Este escenario se muestra en la Figura 3

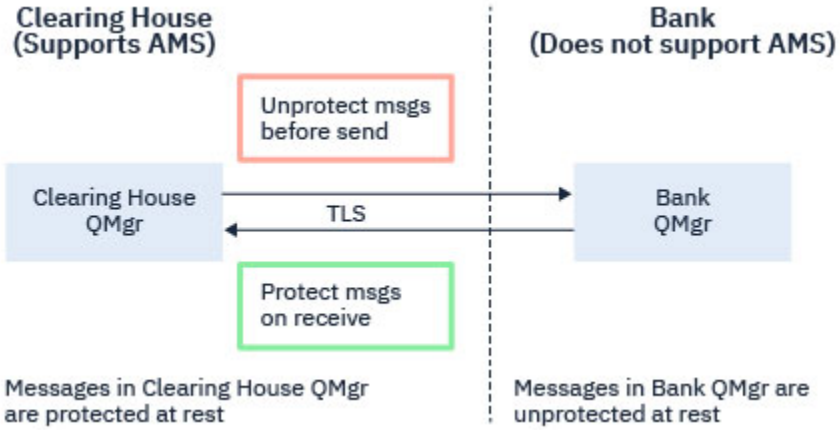


Figura 34. Flujo de mensajes entre business partners

Tareas relacionadas

Configuraciones de ejemplo de intercepción de canal de mensajes de servidor a servidor

z/OS V 9.1.3 ***AMS intercepción en canales de mensajes de servidor a servidor***

La intercepción de canal de mensajes de servidor a servidor proporciona una forma de controlar si los mensajes deben tener políticas de Advanced Message Security (AMS) aplicables, cuando los agentes de canal de mensajes de tipo emisor obtienen mensajes de las colas de transmisión, y los agentes de canal de mensajes de tipo receptor colocan mensajes en las colas de destino.

Esto permite habilitar la protección de AMS en un gestor de colas cuando se comunica mediante los canales de mensajes de servidor a servidor de tipo emisor, servidor, receptor y peticionario con un gestor de colas que no tiene habilitado AMS.

Es decir, los mensajes protegidos de AMS en gestores de colas habilitados para AMS se pueden desproteger antes de ser enviados a gestores de colas no habilitados para AMS, y los mensajes no

protegidos recibidos de gestores de colas no habilitados para AMS se pueden proteger, mediante las políticas de AMS aplicables, en gestores de colas habilitados para AMS.

Configuración de la intercepción de canal de mensajes de servidor a servidor

La intercepción de canal de mensajes de servidor a servidor se configura con el atributo SPLPROT en los canales con un tipo de canal emisor, servidor, receptor o peticionario. Las opciones disponibles para configurar el comportamiento dependen del tipo de canal especificado:

PASSTHRU
Pasa por los mensajes enviados o recibidos por el agente de canal de mensajes de este canal, sin modificarlos.

Este valor es válido para canales con un tipo de canal (**CHLTYPE**) de SDR, RVS, RCVR o RQSTR, y es el valor predeterminado.

REMOVE
Elimine cualquier protección de AMS de los mensajes recuperados de la cola de transmisión por el agente de canal de mensajes y envíe los mensajes al socio.

Cuando el agente de canal de mensajes obtiene un mensaje de la cola de transmisión, si se ha definido una política de AMS para la cola de transmisión, se aplica para eliminar cualquier protección de AMS del mensaje antes de enviar el mensaje a través del canal. Si no se ha definido una política AMS para la cola de transmisión, el mensaje se enviará tal cual.

Este valor solo es válido para canales con un tipo de canal de SDR o SVR.

ASPOLICY
Basándose en la política definida para la cola de destino, aplique la protección de AMS a los mensajes de entrada antes de colocarlos en la cola de destino.

Cuando el agente de canal de mensajes recibe un mensaje de entrada, si se ha definido una política de AMS para la cola de destino, la protección de AMS se aplica al mensaje antes de que el mensaje se coloque en la cola de destino. Si no se ha definido una política de AMS para la cola de destino, el mensaje se coloca en la cola de destino tal como está.

Este valor solo es válido para canales con un tipo de canal de RCVR o RQSTR.

ID de usuario para la interceptación de canal de mensajes

El requisito para los ID de usuario utilizados con la intercepción de canal de mensajes de servidor a servidor es el mismo que el de las aplicaciones habilitadas para AMS existentes. Para un canal en ejecución, el agente de canal de mensajes de envío obtiene mensajes de una cola de transmisión y el agente de canal de mensajes de recepción coloca mensajes en las colas de destino. El campo de ID de usuario de agente de canal de mensajes (MCAUSER), establecido en los canales de servidor a servidor, define el ID de usuario bajo los cuales los agentes de canal de mensajes realizan las solicitudes de colocación y obtención.

Con la intercepción de canal de mensajes de servidor a servidor, las funciones de AMS se realizan durante las solicitudes de obtención y colocación, al igual que con otras aplicaciones habilitadas para AMS. Por lo tanto, los ID de usuario de agente de canal de mensajes tienen los mismos requisitos que los de los ID de usuario de aplicación de AMS.

El MCAUSER utilizado para realizar la obtención y colocación es configurable, y depende de si se trata de un canal de salida o de entrada. Consulte MCAUSER para obtener detalles sobre cómo el ID de usuario elegido realiza acciones en el agente de canal de mensajes. Por lo tanto, el ID de usuario con el que se ejecuta el iniciador de canal es el ID de usuario que se va a utilizar para las funciones de AMS realizadas durante la intercepción de canal de mensajes de servidor a servidor. Por lo tanto, estos ID de usuario tienen los mismos requisitos que los de los ID de usuario de aplicación de AMS.

La autenticación se realiza utilizando las reglas existentes para el canal detallado para los canales con la configuración PUTAUT. Consulte Identificadores de usuario utilizados por el iniciador de canal para obtener más información.

Nota: La intercepción de canal de mensajes de servidor a servidor no tiene en cuenta el valor del atributo de canal PUTAUT.

Tamaño de mensaje y MAXMSGL

Debido a la protección de AMS, el tamaño de los mensajes protegidos será mayor que el tamaño de los mensajes originales.

Los mensajes protegidos son más grandes que los mensajes no protegidos. Por lo tanto, es posible que sea necesario modificar el valor del atributo **MAXMSGL**, en colas y canales, para tener en cuenta el tamaño de los mensajes protegidos.

Referencia relacionada

Configuraciones de ejemplo de intercepción de canal de mensajes de servidor a servidor

Tratamiento de errores

IBM MQ Advanced Message Security define una cola de tratamiento de errores para gestionar los mensajes que contienen errores o los mensajes que no se pueden desproteger.

Los mensajes defectuosos se tratan como casos excepcionales. Si un mensaje recibido no cumple los requisitos de seguridad de la cola en la que se encuentra, por ejemplo, si el mensaje está firmado cuando debería estar cifrado, o si fallan el descifrado o l verificación de la firma, el mensaje se envía a la cola de tratamiento de errores. Un mensaje se puede enviar a la cola de tratamiento de errores por las siguientes razones:

- Discrepancia de calidad de protección: existe una discrepancia en la calidad de protección (QOP) entre el mensaje recibido y la definición QOP de la política de seguridad.
- Error de descifrado - el mensaje no puede descrifrarse.
- Error de cabecera PDMQ - no se puede acceder a la cabecera de mensaje de Advanced Message Security (AMS).
- Discrepancia de tamaños - la longitud de un mensaje tras el descifrado es distinta de la esperada.
- Discrepancia de fuerza del algoritmo de cifrado: el algoritmo de cifrado del mensaje no tiene la fuerza necesaria.
- Error desconocido: se ha producido un error inesperado.

AMS utiliza SYSTEM.PROTECTION.ERROR.QUEUE como cola de manejo de errores. Todos los mensajes colocados por IBM MQ AMS en SYSTEM.PROTECTION.ERROR.QUEUE van precedidos de una cabecera MQDLH.

El administrador de IBM MQ también puede definir el SYSTEM.PROTECTION.ERROR.QUEUE como una cola alias que apunta a otra cola.

z/OS V 9.1.3 A partir de IBM MQ 9.1.3, en IBM MQ for z/OS, si está en uso la intercepción del agente de canal de mensajes (MCA) de servidor a servidor:

- Si por una de las razones anteriormente indicadas, IBM MQ AMS mueve los mensajes de la cola de transmisión a la cola de tratamiento de errores, el MCA emisor simplemente procede a procesar el siguiente mensaje disponible en la cola de transmisión.
- En general, se aplican las reglas de canal existentes para:
  - Colocar mensajes en la cola de mensajes no entregados
  - Acciones realizadas en caso de que fallara la colocación en la cola de mensajes no entregados.

Consulte “Mensajes no entregados para AMS en z/OS” en la página 579 para obtener más información sobre casos específicos.

z/OS V 9.1.3 *Mensajes no entregados para AMS en z/OS*

Escenarios específicos relacionados con la interceptación del agente de canal de mensajes de servidor a servidor en IBM MQ for z/OS.

A partir de IBM MQ 9.1.3, en IBM MQ for z/OS, si está en uso la intercepción del agente de canal de mensajes (MCA) de servidor a servidor:

- Si, después de haber obtenido un mensaje no protegido, el MCA emisor no puede entregar un mensaje por alguna razón, por ejemplo, porque el mensaje es demasiado grande para el canal, si el atributo de canal emisor USEDLQ se establece en YES, el MCA emisor traslada el mensaje a la cola de mensajes no entregados local (DLQ).

  Si SYSTEM.DEAD.LETTER.QUEUE se está utilizando como la cola de mensajes no entregados local, el mensaje se colocará sin protección.

  Nota: IBM MQ AMS no admite la protección de mensajes puestos en las colas de sistema.

  Si se utiliza una cola de mensajes no entregados con nombre como la cola de mensajes no entregados local, el mensaje se colocará protegido si ha definido una política de IBM MQ AMS con el mismo nombre que la cola de mensajes no entregados nombrada y se colocará desprotegida si no ha definido una política adecuada.
- Si no se puede poner un mensaje en la cola de mensajes no entregados local por algún motivo, si la NPMSPEED del canal se establece en NORMAL o el mensaje es permanente, se restituye el lote actual de mensajes y el canal se pone en estado RETRY (Reintentar). De lo contrario, el mensaje se descarta y el agente de canal de mensajes emisor continúa procesando el siguiente mensaje en la cola de transmisión.
- Dado que las políticas de seguridad no tienen ningún efecto sobre la cola SYSTEM.DEAD.LETTER.QUEUE o las otras colas SYSTEM enumeradas en “Protección de colas del sistema en AMS” en la página 652, si SYSTEM.DEAD.LETTER.QUEUE está en uso, los mensajes puestos en esta cola por los agentes de canal de mensajes se colocan tal cual están. Es decir, si los mensajes se han protegido anteriormente, se colocan protegidos; de lo contrario, se colocan desprotegidos.

  Si el atributo DEADQ del gestor de colas se ha establecido en el nombre de una cola de mensajes no entregados alternativa (no del sistema) y no existe una política de AMS con el mismo nombre, los mensajes colocados en esta cola por los agentes de canal de mensajes se colocan tal cual están. Es decir, si los mensajes se han protegido anteriormente, se colocan protegidos; de lo contrario, se colocan desprotegidos.

  Si el atributo DEADQ del gestor de colas se ha establecido en el nombre de una cola de mensajes no entregados alternativa (no del sistema) y existe una política de AMS con el mismo nombre, los mensajes colocados en esta cola por los agentes de canal de mensajes se colocan tal cual están. Si el mensaje ya se ha protegido, no se vuelve a proteger; esto es para evitar una doble protección. Si no existe una política de servicios de método de acceso con el mismo nombre, los mensajes se colocan tal cual.
- Si hay una política para la cola de mensajes no entregados con la opción de tolerancia en el conjunto de mandatos setmqspl establecida en desactivada, es decir '**-t** O', la colocación en cola de mensajes no entregados falla si el mensaje no está protegido por AMS y, por lo tanto, no tiene una cabecera PDMQ. Esto sucede si el mensaje llega al receptor sin una cabecera PDMQ. Es el putter original del mensaje que no tenía una política para el destino y el receptor no tiene establecido SPLPROT (ASPOLICY).
- Es posible que un MCA no pueda colocar un mensaje en la cola de mensajes no entregados, si la política de AMS definida para la cola de mensajes no entregados no permite el ID de usuario con el que se ejecuta el iniciador de canal para proteger el mensaje.
- Los canales receptores suelen colocar mensajes no entregados en la cola de mensajes no entregados local, mientras que los canales emisores suelen colocar los mensajes que no se pueden procesar por alguna razón, por ejemplo, los mensajes demasiado grandes para la cola, o una cabecera MQXQH incorrecta, etc. en la cola de mensajes no entregados local.
- Los manejadores de cola de mensajes no entregados normalmente solo buscan en la cabecera de cola de mensajes no entregados (DLH) y no en la carga útil de mensaje en sí. Por lo tanto, el hecho de que la carga útil de mensaje pueda estar protegida, no impide que los manejadores determinen por qué se ha colocado el mensaje en la cola de mensajes no entregados.
- Si no se ha definido una cola de mensajes no entregados, el canal:
  - Finaliza de forma anómala (y entra en estado de reintento) si no se puede entregar un mensaje permanente.

– Descarta un mensaje no entregado no permanente y continúa ejecutándose.

Conceptos relacionados

"Tratamiento de errores" en la página 579
IBM MQ Advanced Message Security define una cola de tratamiento de errores para gestionar los mensajes que contienen errores o los mensajes que no se pueden desproteger.

Escenarios de usuario

Conozca los posibles escenarios para comprender cuáles son los objetivos empresariales que se pueden alcanzar con Advanced Message Security.

Windows *Guía de inicio rápido para AMS en plataformas Windows*

Use esta guía para configurar rápidamente Advanced Message Security para proporcionar seguridad de mensajes en plataformas Windows. La guía describe cómo crear una base de datos para verificar las identidades de usuario y definir políticas de firma/cifrado para el gestor de colas.

Antes de empezar

Como mínimo, es necesario tener instaladas en el sistema las siguientes características:

- Servidor
- Kit de herramientas de desarrollo (para los programas de ejemplo)
- Advanced Message Security

Consulte las características de IBM MQ para sistemas Windows para obtener información detallada.

Para obtener información sobre cómo utilizar el mandato **setmqenv** para inicializar el entorno actual para que el sistema operativo pueda localizar y ejecutar los mandatos IBM MQ adecuados, consulte setmqenv (set IBM MQ environment).

1. Crear un gestor de colas y una cola

Acerca de esta tarea

Todos los ejemplos siguientes utilizan una cola denominada TEST.Q para pasar mensajes entre aplicaciones. Advanced Message Security utiliza interceptores para firmar y cifrar mensajes en el momento en que los mensajes entran en la infraestructura de IBM MQ a través de la interfaz estándar de IBM MQ. La configuración básica se realiza en IBM MQ y se define en los pasos siguientes.

Puede utilizar IBM MQ Explorer para crear el gestor de colas QM_VERIFY_AMS y su cola local denominada TEST.Q utilizando todos los valores predeterminados del asistente, o puede utilizar los mandatos que se encuentran en `C:\Archivos de programa\IBM\MQ\bin`. Recuerde que debe ser miembro del grupo de usuarios `mqm` para ejecutar los siguientes mandatos administrativos.

Procedimiento

1. Crear un gestor de colas

```
crtmqm QM_VERIFY_AMS
```

2. Inicie el gestor de colas

```
strmqm QM_VERIFY_AMS
```

3. Cree una cola denominada TEST.Q especificando el siguiente mandato en **runmqsc** para el gestor de colas QM_VERIFY_AMS

```
DEFINE QLOCAL(TEST.Q)
```

Resultados

Si se completa el procedimiento, el mandato introducido en **runmqsc** mostrará detalles sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Crear y autorizar usuarios

Acerca de esta tarea

En este ejemplo existen dos usuarios: `alice`, el emisor, y bob, el receptor. Para utilizar la cola de aplicación, estos usuarios deben tener autorización para utilizarla. Asimismo, para utilizar correctamente las políticas de protección que vamos a definir, estos usuarios deben tener acceso a algunas colas del sistema. Para obtener más información sobre el mandato **setmqaut** , consulte **setmqaut**.

Procedimiento

1. Cree los dos usuarios y asegúrese de que HOMEPATH y HOMEDRIVE se hayan establecido para estos usuarios.
2. Autorice a los usuarios para conectarse con el gestor de colas y para trabajar con la cola

   ```
   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. También debe permitir que los dos usuarios examinen la cola de políticas del sistema y coloquen mensajes en la cola de errores.

   ```
   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
   ```

   Atención: IBM MQ optimiza el rendimiento almacenando en memoria caché las políticas para que no tenga que examinar los registros para obtener detalles de política en el SYSTEM.PROTECTION.POLICY.QUEUE en todos los casos.

   IBM MQ no almacena en memoria caché todas las políticas disponibles. Si hay un número alto de políticas, IBM MQ almacena en memoria caché un número limitado de políticas. Por lo tanto, si el gestor de colas tiene un número bajo de políticas definidas, no es necesario proporcionar la opción de examinar a SYSTEM.PROTECTION.POLICY.QUEUE.

   Sin embargo, debe otorgar autorización para examinar esta cola, en caso de que haya un gran número de políticas definidas, o si está utilizando clientes antiguos. El sistema SYSTEM.PROTECTION.ERROR.QUEUE se utiliza para colocar mensajes de error generados por el código AMS. La autorización de colocación sobre esta cola sólo se comprueba cuando se intenta colocar un mensaje de error en la cola. La autorización de colocación sobre la cola no se comprueba cuando intenta transferir u obtener un mensaje de una cola protegida por AMS.

Resultados

Se crean los usuarios y se les otorgan las autorizaciones necesarias.

Qué hacer a continuación

Para verificar si los pasos se han realizado correctamente, utilice los ejemplos `amqsput` y `amqsget` tal como se describe en la sección “7. Probar la configuración” en la página 586.

3. Crear la base de datos de claves y certificados

Acerca de esta tarea

El interceptor necesita la clave pública de los usuarios de los usuarios emisores para cifrar el mensaje. Por lo tanto, se deben crear la base de datos de claves de identidades de usuario que están correlacionadas con claves públicas y privadas. En el sistema real, donde los usuarios y las aplicaciones están distribuidos en varios sistemas, cada usuario tendría su propio almacén de claves privado. De forma similar, en esta guía, creamos bases de datos de claves para `alice` y bob y compartimos los certificados de usuario entre ellos.

Nota: En esta guía, utilizamos aplicaciones de ejemplo escritas en C que se conectan mediante enlaces locales. Si tiene previsto utilizar aplicaciones Java utilizando enlaces de cliente, debe crear un almacén de claves de JKS y certificados mediante el mandato **`keytool`**, que forma parte del JRE (consulte "Guía de inicio rápido para AMS con clientes Java" en la página 603 para obtener más detalles). Para los demás lenguajes, y para las aplicaciones Javaque utilizan enlaces locales, los pasos de esta guía son correctos.

Procedimiento

1. Utilice la GUI de IBM Key Management (`strmqikm.exe`) para crear una nueva base de datos de claves para el usuario `alice`.

   ```
   Type:       CMS
   Filename:   alicekey.kdb
   Location:   C:/Documents and Settings/alice/AMS
   ```

   Nota:

   - Se recomienda utilizar una contraseña fuerte para proteger la base de datos.
   - Asegúrese de que esté seleccionado el recuadro **Ocultar contraseña en un archivo**.
2. Cambie la vista de contenido de la base de datos de claves a **Certificados personales**.
3. Seleccione **New Self Signed**. En este caso de ejemplo se utilizan certificados autofirmados.
4. Cree un certificado que identifique al usuario `alice` para utilizarlo en el cifrado mediante los siguientes campos:

   ```
   Key label: Alice_Cert
   Common Name: alice
   Organisation: IBM
   Country: GB
   ```

   Nota:

   - A los efectos de esta guía, utilizamos un certificado autofirmado que se puede crear sin utilizar una entidad emisora de certificados. Para los sistemas de producción, se recomienda no utilizar certificados autofirmados, sino certificados firmados por una entidad emisora de certificados.
   - El parámetro **`Key label`** especifica el nombre del certificado, que los interceptores consultarán para obtener información necesaria.
   - El parámetro **`Common Name`** especifica los detalles del **Nombre distinguido** (DN), que debe ser exclusivo para cada usuario.
5. Repita los pasos del 1 al 4 para el usuario bob

Resultados

Los dos usuarios `alice` y bob tienen cada uno un certificado autofirmado.

4. Crear keystore.conf

Acerca de esta tarea

Debe apuntar los interceptores de Advanced Message Security al directorio donde se encuentran las bases de datos de claves y los certificados located.This se realiza a través del archivo `keystore.conf`, que contiene dicha información en formato de texto sin formato. Cada usuario debe tener un archivo `keystore.conf` independiente en la carpeta `.mqs`. Este paso debe realizarse tanto para `alice` como para bob.

El contenido de `keystore.conf` debe tener este formato:

```
cms.keystore = dir/keystore_file
cms.certificate = certificate_label
```

Ejemplo

Para este caso de ejemplo, el contenido de `keystore.conf` será el siguiente:

```
cms.keystore = C:/Documents and Settings/alice/AMS/alicekey
cms.certificate = Alice_Cert
```

Nota:

- La vía de acceso del archivo de almacén de claves se debe especificar sin ninguna extensión de archivo.
- La etiqueta del certificado puede incluir espacios, por lo que "Alice_Cert" y "Alice_Cert" (con un espacio al final) por ejemplo, se reconocen como etiquetas de dos certificados diferentes. Sin embargo, para evitar confusiones, es mejor no utilizar espacios en el nombre de la etiqueta.
- Existen los siguientes formatos de almacén de claves: CMS (Cryptographic Message Syntax), JKS (Java Keystore) y JCEKS (Java Cryptographic Extension Keystore). Para obtener más información, consulte “Estructura del archivo de configuración del almacén de claves (keystore.conf) para AMS” en la página 617.
- *%HOMEDRIVE%\%HOMEPATH%*\`.mqs\keystore.conf` (p.ej., C:\Documents and Settings\alice\.mqs\keystore.conf) es la ubicación predeterminada donde Advanced Message Security busca el archivo `keystore.conf`. Para obtener información sobre cómo utilizar una ubicación no predeterminada para `keystore.conf`, consulte “Utilización de almacenes de claves y certificados” en la página 616.
- Para crear el directorio `.mqs` debe usar el indicador de mandatos.

5. Compartir certificados

Acerca de esta tarea

Comparta los certificados entre las dos bases de datos para que cada usuario pueda identificar correctamente al otro. Esto se realiza extrayendo el certificado público de cada usuario en un archivo, que después se añade a la base de datos de claves del otro usuario.

Nota: Tenga cuidado y utilice la opción *extraer* y no la opción *exportar*. *Extraer* obtiene la clave pública del usuario, mientras que *exportar* obtiene ambas claves, la pública y la privada. El uso de *exportar* por error comprometería por completo la aplicación, pasando su clave privada.

Procedimiento

1. Extraiga el certificado que identifica a `alice` a un archivo externo.

   ```
   runmqakm -cert -extract -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
   -label Alice_Cert -target alice_public.arm
   ```

2. Añada el certificado al almacén de clavos `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 los pasos para bob:

```
runmqakm -cert -extract -db "C:/Documents and Settings/bob/AMS/bobkey.kdb" -pw passw0rd
-label Bob_Cert -target bob_public.arm

runmqakm -cert -add -db "C:/Documents and Settings/alice/AMS/alicekey.kdb" -pw passw0rd
-label Bob_Cert -file bob_public.arm
```

Resultados

Los dos usuarios `alice` y bob pueden identificar ahora correctamente al otro mediante la creación y compartición de certificados autofirmados.

Qué hacer a continuación

Para verificar que un certificado está en el almacén de claves, examínelo utilizando la GUI o ejecute los siguientes mandatos que imprimen los detalles:

```
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. Definir la política de cola

Acerca de esta tarea

Después de crear el gestor de colas y preparar interceptores para interceptar mensajes y acceder a claves de cifrado, podemos comenzar a definir políticas de protección en `QM_VERIFY_AMS` mediante el mandato `setmqspl`. Consulte setmqspl para obtener más información sobre este mandato. Cada nombre de política debe ser el mismo que el nombre de cola al que se debe aplicar.

Ejemplo

Este es un ejemplo de una política definida para la cola `TEST.Q`. En el ejemplo, los mensajes se firman con el algoritmo SHA1 y se cifran con el algoritmo AES256. `alice` es el único emisor válido y bob es el único receptor de los mensajes en esta cola:

```
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: Los DN coinciden exactamente con los especificados en el certificado del usuario respectivo de la base de datos de claves.

Qué hacer a continuación

Para verificar la política que ha definido, emita el mandato siguiente:

```
dspmqspl -m QM_VERIFY_AMS
```

Para imprimir los detalles de la política como un conjunto de mandatos `setmqspl`, utilice el distintivo `-export`. Esto permite almacenar políticas ya definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Probar la configuración

Acerca de esta tarea

Puede ejecutar programas diferentes bajo usuarios diferentes para verificar si la aplicación se ha configurado debidamente.

Procedimiento

1. Cambie el usuario de modo que se ejecute como usuario `alice`

   Pulse con el botón derecho del ratón en `cmd.exe` y seleccione **Ejecutar como...**. Cuando se le solicite, inicie una sesión como el usuario `alice`.
2. Como usuario `alice`, coloque un mensaje utilizando una aplicación de ejemplo:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```
3. Escriba el texto del mensaje y pulse Intro.
4. Cambie el usuario de modo que se ejecute como usuario bob

   Pulse con el botón derecho del ratón en `cmd.exe` y seleccione **Ejecutar como...** para abrir otra ventana. Cuando se le solicite, inicie una sesión como el usuario bob.
5. Como usuario bob, obtenga un mensaje utilizando una aplicación de ejemplo:

   ```
   amqsget TEST.Q QM_VERIFY_AMS
   ```

Resultados

Si la aplicación se ha configurado debidamente para ambos usuarios, el mensaje del usuario `alice` se visualiza cuando bob ejecuta la aplicación de obtención.

8. Probar el cifrado

Acerca de esta tarea

Para verificar que el cifrado se realiza según lo esperado, cree una cola de alias que haga referencia a la cola original `TEST.Q`. Esta cola de alias no tendrá ninguna política de seguridad, por lo que ningún usuario tendrá la información para descifrar el mensaje y, por lo tanto, se mostrarán los datos cifrados.

Procedimiento

1. Utilizando el mandato **runmqsc** en el gestor de colas QM_VERIFY_AMS, cree una cola de alias.

   ```
   DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
   ```
2. Otorgue a bob el acceso para examinar desde la cola de alias

   ```
   setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
   ```
3. Como usuario `alice`, coloque otro mensaje utilizando una aplicación de ejemplo al igual que antes:

   ```
   amqsput TEST.Q QM_VERIFY_AMS
   ```
4. Como usuario bob, examine el mensaje utilizando una aplicación de ejemplo utilizando la cola de alias esta vez:

   ```
   amqsbcg TEST.ALIAS QM_VERIFY_AMS
   ```
5. Como usuario bob, obtenga el mensaje utilizando una aplicación de ejemplo desde la cola local:

```
amqsget TEST.Q QM_VERIFY_AMS
```

Resultados

La salida de la aplicación amqsbcg muestra los datos cifrados que hay en la cola, lo que demuestra que el mensaje se ha cifrado.

UNIX *Guía de inicio rápido para AMS en UNIX*

Use esta guía para configurar rápidamente Advanced Message Security para proporcionar seguridad de mensajes en UNIX. La guía describe cómo crear una base de datos para verificar las identidades de usuario y definir políticas de firma/cifrado para el gestor de colas.

Antes de empezar

Como mínimo, es necesario tener instalados en el sistema los siguientes componentes:

- Tiempo de ejecución
- Servidor
- Programas de ejemplo
- IBM Kit de seguridad global
- Advanced Message Security

Consulte los temas siguientes para ver los nombres de componentes en cada plataforma específica:

- Linux Componentes de IBM MQ para sistemas Linux
- AIX Componentes de IBM MQ para sistemas AIX
- Solaris Componentes de IBM MQ para sistemas Solaris

1. Crear un gestor de colas y una cola

Acerca de esta tarea

Todos los ejemplos siguientes utilizan una cola denominada TEST.Q para pasar mensajes entre aplicaciones. Advanced Message Security utiliza interceptores para firmar y cifrar mensajes en el momento en que los mensajes entran en la infraestructura de IBM MQ a través de la interfaz estándar de IBM MQ. La configuración básica se realiza en IBM MQ y se define en los pasos siguientes.

Puede utilizar IBM MQ Explorer para crear el gestor de colas QM_VERIFY_AMS y su cola local denominada TEST.Q utilizando todos los valores predeterminados del asistente, o puede utilizar los mandatos que se encuentran en *MQ_INSTALLATION_PATH*/bin. Recuerde que debe ser miembro del grupo de usuarios mqm para ejecutar los siguientes mandatos administrativos.

Procedimiento

1. Crear un gestor de colas

```
crtmqm QM_VERIFY_AMS
```

2. Inicie el gestor de colas

```
strmqm QM_VERIFY_AMS
```

3. Cree una cola denominada TEST.Q especificando el siguiente mandato en **runmqsc** para el gestor de colas QM_VERIFY_AMS

```
DEFINE QLOCAL(TEST.Q)
```

Resultados

Si el procedimiento se ha ejecutado correctamente, el siguiente mandato especificado en **runmqsc** mostrará detalles sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Crear y autorizar usuarios

Acerca de esta tarea

En este ejemplo existen dos usuarios: alice, el emisor, y bob, el receptor. Para utilizar la cola de aplicación, estos usuarios deben tener autorización para utilizarla. Asimismo, para utilizar correctamente las políticas de protección que vamos a definir, estos usuarios deben tener acceso a algunas colas del sistema. Para obtener más información sobre el mandato **setmqaut** , consulte **setmqaut**.

Procedimiento

1. Cree los dos usuarios.

```
useradd alice
useradd bob
```

2. Autorice a los usuarios para conectarse con el gestor de colas y para trabajar con la cola

```
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. También debe permitir que los dos usuarios examinen la cola de políticas del sistema y coloquen mensajes en la cola de errores.

```
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
```

Atención: IBM MQ optimiza el rendimiento almacenando en memoria caché las políticas para que no tenga que examinar los registros para obtener detalles de política en el SYSTEM.PROTECTION.POLICY.QUEUE en todos los casos.

IBM MQ no almacena en memoria caché todas las políticas disponibles. Si hay un número alto de políticas, IBM MQ almacena en memoria caché un número limitado de políticas. Por lo tanto, si el gestor de colas tiene un número bajo de políticas definidas, no es necesario proporcionar la opción de examinar a SYSTEM.PROTECTION.POLICY.QUEUE.

Sin embargo, debe otorgar autorización para examinar esta cola, en caso de que haya un gran número de políticas definidas, o si está utilizando clientes antiguos. El sistema SYSTEM.PROTECTION.ERROR.QUEUE se utiliza para colocar mensajes de error generados por el código AMS. La autorización de colocación sobre esta cola sólo se comprueba cuando se intenta colocar un mensaje de error en la cola. La autorización de colocación sobre la cola no se comprueba cuando intenta transferir u obtener un mensaje de una cola protegida por AMS.

Resultados

Se crean los grupos de usuarios y se les otorgan las autorizaciones necesarias. De este forma, los usuarios que se asignen a esos grupos también tendrán permisos para conectarse al gestor de colas y realizar operaciones put y get con la cola.

Qué hacer a continuación

Para verificar si los pasos se han realizado correctamente, utilice los ejemplos `amqsput` y `amqsget` tal como se describe en la sección “8. Probar el cifrado” en la página 592.

3. Crear la base de datos de claves y certificados

Acerca de esta tarea

Para cifrar el mensaje, el interceptor necesita la clave privada del usuario emisor y las claves públicas del destinatario o los destinatarios. Por lo tanto, se deben crear la base de datos de claves de identidades de usuario que están correlacionadas con claves públicas y privadas. En el sistema real, donde los usuarios y las aplicaciones están distribuidos en varios sistemas, cada usuario tendría su propio almacén de claves privado. De forma similar, en esta guía, creamos bases de datos de claves para `alice` y bob y compartimos los certificados de usuario entre ellos.

Nota: En esta guía, utilizamos aplicaciones de ejemplo escritas en C que se conectan mediante enlaces locales. Si tiene previsto utilizar aplicaciones Java utilizando enlaces de cliente, debe crear un almacén de claves de JKS y certificados mediante el mandato **keytool**, que forma parte del JRE (consulte “Guía de inicio rápido para AMS con clientes Java” en la página 603 para obtener más detalles). Para los demás lenguajes, y para las aplicaciones Javaque utilizan enlaces locales, los pasos de esta guía son correctos.

Procedimiento

1. Cree una nueva base de datos de claves para el usuario `alice`

   ```
   mkdir /home/alice/.mqs -p
   runmqakm -keydb -create -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -stash
   ```

   Nota:

   - Se recomienda utilizar una contraseña fuerte para proteger la base de datos.
   - El parámetro **stash** almacena la contraseña en el archivo `key.sth`, que los interceptores pueden utilizar para abrir la base de datos.
2. Asegúrese de que la base de datos de claves sea legible.

   ```
   chmod +r /home/alice/.mqs/alicekey.kdb
   ```

3. Cree un certificado que identifique al usuario `alice` para utilizarlo en el cifrado

   ```
   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:

   - A los efectos de esta guía, utilizamos un certificado autofirmado que se puede crear sin utilizar una entidad emisora de certificados. Para los sistemas de producción, se recomienda no utilizar certificados autofirmados, sino certificados firmados por una entidad emisora de certificados.
   - El parámetro **label** especifica el nombre para el certificado, que los interceptores buscarán para recibir información necesaria.
   - El parámetro **DN** especifica los detalles del **Nombre distinguido** (DN), que debe ser exclusivo para cada usuario.
4. Ahora que hemos creado la base de datos de claves, debemos establecer su propiedad y comprobar que sea ilegible para los demás usuarios.

   ```
   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 los pasos del 1 al 4 para el usuario bob

Resultados

Los dos usuarios `alice` y bob tienen cada uno un certificado autofirmado.

4. Crear keystore.conf

Acerca de esta tarea

Debe apuntar los interceptores de Advanced Message Security al directorio donde residen las bases de datos y certificados. Esto se realiza a través del archivo `keystore.conf` , que contiene esa información en formato de texto sin formato. Cada usuario debe tener un archivo `keystore.conf` independiente en la carpeta `.mqs`. Este paso debe realizarse tanto para `alice` como para bob.

El contenido de `keystore.conf` debe tener este formato:

```
cms.keystore = dir/keystore_file
cms.certificate = certificate_label
```

Ejemplo

Para este caso de ejemplo, el contenido de `keystore.conf` será el siguiente:

```
cms.keystore = /home/alice/.mqs/alicekey
cms.certificate = Alice_Cert
```

Nota:

- La vía de acceso del archivo de almacén de claves se debe especificar sin ninguna extensión de archivo.
- Existen los siguientes formatos de almacén de claves: CMS (Cryptographic Message Syntax), JKS (Java Keystore) y JCEKS (Java Cryptographic Extension Keystore). Para obtener más información, consulte “Estructura del archivo de configuración del almacén de claves (keystore.conf) para AMS” en la página 617.
- `HOME/.mqs/keystore.conf` es la ubicación predeterminada donde Advanced Message Security busca el archivo `keystore.conf`. Para obtener información sobre cómo utilizar una ubicación no predeterminada para `keystore.conf`, consulte “Utilización de almacenes de claves y certificados” en la página 616.

5. Compartir certificados

Acerca de esta tarea

Comparta los certificados entre las dos bases de datos para que cada usuario pueda identificar correctamente al otro. Esto se realiza extrayendo el certificado público de cada usuario en un archivo, que después se añade a la base de datos de claves del otro usuario.

Nota: Tenga cuidado y utilice la opción *extraer* y no la opción *exportar*. *Extraer* obtiene la clave pública del usuario, mientras que *exportar* obtiene ambas claves, la pública y la privada. El uso de *exportar* por error comprometería por completo la aplicación, pasando su clave privada.

Procedimiento

1. Extraiga el certificado que identifica a `alice` a un archivo externo.

   ```
   runmqakm -cert -extract -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Alice_Cert
   -target alice_public.arm
   ```

2. Añada el certificado al almacén de clavos `bob's`:

   ```
   runmqakm -cert -add -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Alice_Cert -file
   alice_public.arm
   ```

3. Repita el paso para bob:

```
runmqakm -cert -extract -db /home/bob/.mqs/bobkey.kdb -pw passw0rd -label Bob_Cert -target
bob_public.arm
```

4. Añada el certificado para bob al almacén de claves `alice's`:

```
runmqakm -cert -add -db /home/alice/.mqs/alicekey.kdb -pw passw0rd -label Bob_Cert -file
bob_public.arm
```

Resultados

Los dos usuarios `alice` y bob pueden identificar ahora correctamente al otro mediante la creación y compartición de certificados autofirmados.

Qué hacer a continuación

Verifique que un certificado esté en el almacén de claves ejecutando los siguientes mandatos que imprimen los detalles:

```
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. Definir la política de cola

Acerca de esta tarea

Después de crear el gestor de colas y preparar interceptores para interceptar mensajes y acceder a claves de cifrado, podemos comenzar a definir políticas de protección en `QM_VERIFY_AMS` mediante el mandato `setmqspl`. Consulte setmqspl para obtener más información sobre este mandato. Cada nombre de política debe ser el mismo que el nombre de cola al que se debe aplicar.

Ejemplo

Este es un ejemplo de una política definida para la cola `TEST.Q`. En este ejemplo, el usuario `alice` firma los mensajes utilizando el algoritmo SHA1 y los cifra utilizando el algoritmo AES de 256 bits. `alice` es el único emisor válido y bob es el único receptor de los mensajes en esta cola:

```
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: Los DN coinciden exactamente con los especificados en el certificado del usuario respectivo de la base de datos de claves.

Qué hacer a continuación

Para verificar la política que ha definido, emita el mandato siguiente:

```
dspmqspl -m QM_VERIFY_AMS
```

Para imprimir los detalles de la política como un conjunto de mandatos `setmqspl`, utilice el distintivo `-export`. Esto permite almacenar políticas ya definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Probar la configuración

Acerca de esta tarea

Puede ejecutar programas diferentes bajo usuarios diferentes para verificar si la aplicación se ha configurado debidamente.

Procedimiento

1. Cambie al directorio que contiene los ejemplos. Si MQ está instalado en una ubicación no predeterminada, puede estar en otro lugar.

```
cd /opt/mqm/samp/bin
```

2. Cambie el usuario de modo que se ejecute como usuario `alice`

```
su alice
```

3. Como usuario `alice`, coloque un mensaje utilizando una aplicación de ejemplo:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Escriba el texto del mensaje y pulse Intro.
5. Deje de ejecutarse como el usuario `alice`

```
exit
```

6. Cambie el usuario de modo que se ejecute como usuario bob

```
su bob
```

7. Como usuario bob, obtenga un mensaje utilizando una aplicación de ejemplo:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Resultados

Si la aplicación se ha configurado debidamente para ambos usuarios, el mensaje del usuario `alice` se visualiza cuando bob ejecuta la aplicación de obtención.

8. Probar el cifrado

Acerca de esta tarea

Para verificar que el cifrado se realiza según lo esperado, cree una cola de alias que haga referencia a la cola original TEST.Q. Esta cola de alias no tendrá ninguna política de seguridad, por lo que ningún usuario tendrá la información para descifrar el mensaje y, por lo tanto, se mostrarán los datos cifrados.

Procedimiento

1. Utilizando el mandato **runmqsc** en el gestor de colas QM_VERIFY_AMS, cree una cola de alias.

```
DEFINE QALIAS(TEST.ALIAS) TARGET(TEST.Q)
```

2. Otorgue a bob el acceso para examinar desde la cola de alias

```
setmqaut -m QM_VERIFY_AMS -n TEST.ALIAS -t queue -p bob +browse
```

3. Como usuario `alice`, coloque otro mensaje utilizando una aplicación de ejemplo al igual que antes:

```
./amqsput TEST.Q QM_VERIFY_AMS
```

4. Como usuario bob, examine el mensaje utilizando una aplicación de ejemplo utilizando la cola de alias esta vez:

```
./amqsbcg TEST.ALIAS QM_VERIFY_AMS
```

5. Como usuario bob, obtenga el mensaje utilizando una aplicación de ejemplo desde la cola local:

```
./amqsget TEST.Q QM_VERIFY_AMS
```

Resultados

La salida de la aplicación `amqsbcg` mostrará los datos cifrados que hay en la cola, lo que demuestra que el mensaje se ha cifrado.

z/OS *Configuraciones de ejemplo en z/OS*

En esta sección se proporciona configuraciones de políticas de ejemplo y certificados para los escenarios de gestión de colas de Advanced Message Security en z/OS.

Consulte Configuración de Advanced Message Security for z/OS para obtener información detallada sobre cómo configurar Advanced Message Security.

Los ejemplos describen las políticas de Advanced Message Security necesarias y los certificados digitales que deben existir en relación con los usuarios y los conjuntos de claves. Los ejemplos presuponen que los usuarios implicados en los escenarios se han configurado siguiendo las instrucciones proporcionadas en Otorgar a usuarios permisos de recursos para Advanced Message Security.

V 9.1.3 Además, a partir de IBM MQ 9.1.3, consulte Ejemplos de intercepción del canal de mensajes de servidor a servidor.

z/OS *Gestión de colas locales de mensajes protegidos por integridad en z/OS*

En este ejemplo se describen las políticas de Advanced Message Security y los certificados necesarios para enviar y recuperar los mensajes protegidos por integridad a y desde una cola local para las aplicaciones de transferencia y obtención.

El gestor de colas de ejemplo y la cola son:

```
BNK6        - Queue manager
FIN.XFER.Q7 - Local queue
```

Se utilizan estos usuarios:

```
WMQBNK6  - AMS task user
TELLER5  - Sending user
FINADM2  - Recipient user
```

Crear el certificado de usuario

En este ejemplo, solo es necesario un certificado de usuario. Este es el certificado del usuario emisor necesario para firmar mensajes protegidos por integridad. El usuario emisor es 'TELLER5'.

El certificado de la CA (Certificate Authority) también es necesario. El certificado de la CA es el certificado de la autoridad que ha emitido el certificado del usuario. Puede ser una cadena de certificados. Si es así, todos los certificados de la cadena serán necesarios en el conjunto de claves del usuario de la tarea Advanced Message Security, en este caso el usuario WMQBNK6.

Se puede crear un certificado de CA mediante el mandato RACDCERT de RACF. Este certificado se utiliza para emitir los certificados de usuario. Por ejemplo:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Este mandato RACDCERT crea un certificado de CA que se puede utilizar para emitir un certificado de usuario para el usuario 'TELLER5'. Por ejemplo:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

La instalación tendrá procedimientos para seleccionar o crear un certificado de CA, así como procedimientos para emitir los certificados y distribuirlos en los sistemas relevantes.

Cuando se exportan e importan estos certificados, Advanced Message Security requieren:

- El certificado de CA (cadena).
- El certificado de usuario y su clave privada.

Si está utilizando RACF, se puede utilizar el mandato RACDCERT EXPORT para exportar los certificados a un conjunto de datos y se puede utilizar el mandato RACDCERT ADD para importar certificados desde el conjunto de datos. Para obtener información acerca de estos y otros mandatos RACDCERT, consulte *z/OS: Security Server RACF Command Language Reference*.

En este caso, los certificados son necesarios en el sistema z/OS que ejecuta el gestor de colas BNK6.

Cuando se importan los certificados en el sistema z/OS que ejecuta BNK6, el certificado de usuario requiere el atributo TRUST. El mandato RACDCERT ALTER se puede utilizar para añadir el atributo TRUST al certificado. Por ejemplo:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
```

En este ejemplo, no se requiere ningún certificado para el usuario receptor.

Conectar los certificados a los conjuntos de claves relevantes

Una vez creados o importados los certificados necesarios, deben conectarse a los conjuntos de claves de usuario adecuados en el sistema z/OS que ejecuta BNK6. Para crear los conjuntos de claves utilice los mandatos RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario de la tarea Advanced Message Security, WMQBNK6, y un conjunto de claves para el usuario emisor, 'TELLER5'. Tenga en cuenta que el nombre del conjunto de claves drq.ams.keyring es obligatorio y que el nombre distingue entre mayúsculas y minúsculas.

Una vez creados los conjuntos de claves, se pueden conectar los certificados relevantes:

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

El certificado del usuario emisor se debe conectar como DEFAULT. Si el usuario emisor tiene más de un certificado en drq.ams.keyring, se utiliza el certificado predeterminado para fines de firma.

Advanced Message Security no reconoce la creación y modificación de los certificados hasta que se ha detenido y reiniciado o hasta que se emita el mandato z/OS **MODIFY** para renovar la configuración de certificados de Advanced Message Security. Por ejemplo:

```
F BNK6AMSM,REFRESH KEYRING
```

Crear la política de Advanced Message Security

En este ejemplo, los mensajes protegidos por integridad los coloca en la cola FIN.XFER.Q7 una aplicación que se ejecuta como el usuario 'TELLER5' y los recupera de la misma cola una aplicación que se ejecuta como el usuario 'FINADM2', por lo tanto, solo se requiere una política de Advanced Message Security.

El programa de utilidad política de seguridad de mensaje (CSQ0UTIL) Advanced Message Security se crean utilizando el programa de utilidad CSQ0UTIL que se describe en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).

Utilice el programa de utilidad CSQ0UTIL para ejecutar los mandatos siguientes:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK6. El nombre de política y la cola asociada es FIN.XFER.Q7. El algoritmo que se utiliza para generar la firma del emisor es MD5 y el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US'.

Después de definir la política, reinicie el gestor de colas BNK6, o utilice el mandato z/OS **MODIFY** para renovar la configuración de políticas de Advanced Message Security. Por ejemplo:

```
F BNK6AMSM,REFRESH POLICY
```

z/OS *Gestión de colas locales de mensajes protegidos por privacidad en z/OS*

En este ejemplo se describen las políticas de Advanced Message Security y los certificados necesarios para enviar y recuperar los mensajes protegidos por privacidad a y desde una cola local para las aplicaciones de transferencia y obtención. Los mensajes protegidos por privacidad están firmados y cifrados.

El gestor de colas de ejemplo y el gestor local son los siguientes:

```
BNK6        - Queue manager
FIN.XFER.Q8 - Local queue
```

Se utilizan estos usuarios:

```
WMQBNK6  - AMS task user
TELLER5  - Sending user
FINADM2  - Recipient user
```

Los pasos para configurar este escenario son:

Crear el certificado de usuario

En este ejemplo, se necesitan dos certificados de usuario. Estos son el certificado del usuario emisor necesario para firmar mensajes y el certificado del usuario receptor necesario para cifrar y descifrar los datos de mensajes. El usuario emisor es 'TELLER5' y el usuario receptor es 'FINADM2'.

El certificado de la CA (Certificate Authority) también es necesario.El certificado de la CA es el certificado de la autoridad que ha emitido el certificado del usuario. Puede ser una cadena de certificados. Si es así, todos los certificados de la cadena serán necesarios en el conjunto de claves del usuario de la tarea Advanced Message Security, en este caso el usuario WMQBNK6.

Se puede crear un certificado de CA mediante el mandato RACDCERT de RACF. Este certificado se utiliza para emitir los certificados de usuario. Por ejemplo:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Este mandato RACDCERT crea un certificado de CA que se puede utilizar para emitir un certificado de usuario para el usuario 'TELLER5' y el usuario 'FINADM2'. Por ejemplo:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

La instalación tendrá procedimientos para seleccionar o crear un certificado de CA, así como procedimientos para emitir los certificados y distribuirlos en los sistemas relevantes.

Cuando se exportan e importan estos certificados, Advanced Message Security requieren:

- El certificado de CA (cadena).
- El certificado del usuario emisor y su clave privada.
- El certificado del usuario receptor y su clave privada.

Si está utilizando RACF, se puede utilizar el mandato RACDCERT EXPORT para exportar los certificados a un conjunto de datos y se puede utilizar el mandato RACDCERT ADD para importar certificados desde el conjunto de datos. Para obtener más información sobre estos y otros mandatos RACDCERT, consulte RACDCERT (Gestionar certificados digitales RACF) en la publicación *z/OS: Security Server RACF Command Language Reference*.

En este caso, los certificados son necesarios en el sistema z/OS que ejecuta el gestor de colas BNK6.

Cuando se importan los certificados en el sistema z/OS que ejecuta BNK6, los certificados de usuarios requieren el atributo TRUST. El mandato RACDCERT ALTER se puede utilizar para añadir el atributo TRUST al certificado. Por ejemplo:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Conectar los certificados a los conjuntos de claves relevantes

Una vez creados o importados los certificados necesarios, deben conectarse a los conjuntos de claves de usuario adecuados en el sistema z/OS que ejecuta BNK6. Para crear los conjuntos de claves utilice el mandato RACDCERT ADDRING:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario de la tarea Advanced Message Security y un conjunto de claves para el usuario emisor y el usuario receptor. Tenga en cuenta que el nombre del conjunto de claves drq.ams.keyring es obligatorio y que el nombre distingue entre mayúsculas y minúsculas.

Una vez creados los conjuntos de claves, se pueden conectar los certificados relevantes.

```
RACDCERT ID(WMQBNK6) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
```

```
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Los certificados del usuario emisor y el usuario receptor se deben conectar como DEFAULT. Si el usuario tiene más de un certificado en drq.ams.keyring, se utiliza el certificado predeterminado para fines de firma y descifrado.

El certificado del usuario emisor también debe estar conectado al conjunto de claves del usuario de la tarea Advanced Message Security con USAGE(SITE). Esto es debido a que la tarea Advanced Message Security necesita la clave pública del receptor cuando cifra los datos de mensajes. USAGE(SITE) impide que se pueda acceder a la clave privada desde el conjunto de claves.

Advanced Message Security no reconoce la creación y modificación de los certificados hasta que se ha detenido y reiniciado o hasta que se emita el mandato z/OS **MODIFY** para renovar la configuración de certificados de Advanced Message Security. Por ejemplo:

```
F BNK6AMSM,REFRESH KEYRING
```

Crear la política de Advanced Message Security

En este ejemplo, los mensajes protegidos por privacidad los coloca en la cola FIN.XFER.Q8 una aplicación que se ejecuta como el usuario 'TELLER5' y los recupera de la misma cola una aplicación que se ejecuta como el usuario 'FINADM2', por lo tanto, solo se requiere una política de Advanced Message Security.

El programa de utilidad política de seguridad de mensaje (CSQ0UTIL) Advanced Message Security se crean utilizando el programa de utilidad CSQ0UTIL que se describe en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).

Utilice el programa de utilidad CSQ0UTIL para ejecutar los mandatos siguientes:

```
setmqspl -m BNK6 -p FIN.XFER.Q8 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK6. El nombre de política y la cola asociada es FIN.XFER.Q8. El algoritmo que se utiliza para generar la firma del emisor es SHA1 y el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US' y el nombre del usuario receptor es 'CN=FinAdm2,O=BCO,C=US'. El algoritmo que se utiliza para cifrar datos de mensajes es 3DES.

Después de definir la política, reinicie el gestor de colas BNK6, o utilice el mandato z/OS **MODIFY** para renovar la configuración de políticas de Advanced Message Security. Por ejemplo:

```
F BNK6AMSM,REFRESH POLICY
```

z/OS *Gestión de colas remotas de mensajes protegidos por integridad en z/OS*

En este ejemplo se describen las políticas de Advanced Message Security y los certificados necesarios para enviar y recuperar los mensajes protegidos por integridad a y desde colas gestionadas por dos gestores de colas diferentes. Los dos gestores de colas pueden estar ejecutándose en el mismo sistema z/OS, en sistemas z/OS diferentes o un gestor de colas puede estar en un sistema distribuido que ejecuta Advanced Message Security.

Los gestores de colas de ejemplo y las colas son:

```
BNK6        - Sending queue manager
BNK7        - Recipient queue manager
```

```
FIN.XFER.Q7 - Remote queue on BNK6
FIN.RCPT.Q7 - Local queue on BNK7
```

Nota: En este ejemplo, BNK6 y BNK7 son los gestores de colas que se ejecutan en diferentes sistemas z/OS.

Se utilizan estos usuarios:

```
WMQBNK6  - AMS task user on BNK6
WMQBNK7  - AMStask user on BNK7
TELLER5  - Sending user on BNK6
FINADM2  - Recipient user on BNK7
```

Los pasos para configurar este escenario son:

Crear el certificado de usuario

En este ejemplo, solo es necesario un certificado de usuario. Este es el certificado del usuario emisor necesario para firmar mensajes protegidos por integridad. El usuario emisor es 'TELLER5'.

El certificado de la CA (Certificate Authority) también es necesario.El certificado de la CA es el certificado de la autoridad que ha emitido el certificado del usuario. Puede ser una cadena de certificados. Si es así, todos los certificados de la cadena serán necesarios en el conjunto de claves del usuario de la tarea Advanced Message Security, en este caso el usuario WMQBNK7.

Se puede crear un certificado de CA mediante el mandato RACDCERT de RACF. Este certificado se utiliza para emitir los certificados de usuario. Por ejemplo:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Este mandato RACDCERT crea un certificado de CA que se puede utilizar para emitir un certificado de usuario para el usuario 'TELLER5'. Por ejemplo:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

La instalación tendrá procedimientos para seleccionar o crear un certificado de CA, así como procedimientos para emitir los certificados y distribuirlos en los sistemas relevantes.

Cuando se exportan e importan estos certificados, Advanced Message Security requieren:

- El certificado de CA (cadena).
- El certificado del usuario emisor y su clave privada.

Si está utilizando RACF, se puede utilizar el mandato RACDCERT EXPORT para exportar los certificados a un conjunto de datos y se puede utilizar el mandato RACDCERT ADD para importar certificados desde el conjunto de datos. Para obtener más información sobre estos y otros mandatos RACDCERT, consulte RACDCERT (Gestionar certificados digitales RACF) en la publicación *z/OS: Security Server RACF Command Language Reference*.

En este caso, los certificados son necesarios en el sistema z/OS que ejecuta el gestor de colas BNK6 y BNK7.

En este ejemplo, el certificado de emisor se debe importar al sistema z/OS que ejecuta BNK6 y el certificado de CA se debe importar al sistema z/OS que ejecuta BNK7. Cuando se han importado los certificados de usuario, el certificado de usuario requiere el atributo TRUST. El mandato RACDCERT ALTER se puede utilizar para añadir el atributo TRUST al certificado. Por ejemplo, en BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
```

Conectar los certificados a los conjuntos de claves relevantes

Una vez creados o importados los certificados necesarios, deben conectarse a los conjuntos de claves de usuario adecuados en el sistema z/OS que ejecuta BNK6 y BNK7.

Para crear los conjuntos de claves utilice el mandato RACDCERT ADDRING en BNK6:

```
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario emisor en BNK6. Tenga en cuenta que el nombre del conjunto de claves drq.ams.keyring es obligatorio y que el nombre distingue entre mayúsculas y minúsculas.

En BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario de tarea de Advanced Message Security en BNK7. No se requiere ningún conjunto de claves de usuario para 'TELLER5' en BNK7.

Una vez creados los conjuntos de claves, se pueden conectar los certificados relevantes.

En BNK6:

```
RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

En BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))
```

El certificado del usuario emisor se debe conectar como DEFAULT. Si el usuario emisor tiene más de un certificado en drq.ams.keyring, se utiliza el certificado predeterminado para fines de firma.

Advanced Message Security no reconoce la creación y modificación de los certificados hasta que se ha detenido y reiniciado o hasta que se emita el mandato z/OS **MODIFY** para renovar la configuración de certificados de Advanced Message Security. Por ejemplo:

En BNK6:

```
F BNK6AMSM,REFRESH,KEYRING
```

En BNK7:

```
F BNK7AMSM,REFRESH,KEYRING
```

Crear las políticas de Advanced Message Security

En este ejemplo, los mensajes protegidos por integridad los coloca en la cola FIN.XFER.Q7 en BNK6 una aplicación que se ejecuta como el usuario 'TELLER5' y los recupera de la cola local FIN.RCPT.Q7 en BNK7 una aplicación que se ejecuta como el usuario 'FINADM2' , por lo tanto, se requieren dos políticas de Advanced Message Security.

El programa de utilidad política de seguridad de mensaje (CSQ0UTIL) Advanced Message Security se crean utilizando el programa de utilidad CSQ0UTIL que se describe en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).

Utilice el programa de utilidad CSQ0UTIL para ejecutar el mandato siguiente para definir una política de integridad para la cola remota en BNK6:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK6. El nombre de política y la cola asociada es FIN.XFER.Q7. El algoritmo que se utiliza para generar la firma del emisor es MD5 y el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US'.

Utilice también el programa de utilidad CSQ0UTIL para ejecutar el mandato siguiente para definir una política de integridad para la cola local en BNK7:

```
setmqspl -m BNK7 -p FIN.RCPT.Q7 -s MD5 -a CN=Teller5,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK7. El nombre de política y la cola asociada es FIN.RCPT.Q7. El algoritmo que se espera para la firma del emisor es MD5 y el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US'.

Después de definir dos políticas, reinicie los gestores de colas BNK6 y BNK7, o utilice el mandato z/OS **MODIFY** para renovar las configuraciones de políticas de Advanced Message Security. Por ejemplo:

En BNK6:

```
F BNK6AMSM,REFRESH,POLICY
```

En BNK7:

```
F BNK7AMSM,REFRESH,POLICY
```

z/OS *Gestión de colas remotas de mensajes protegidos por privacidad en z/OS*

En este ejemplo se describen las políticas de Advanced Message Security y los certificados necesarios para enviar y recuperar los mensajes protegidos por privacidad a y desde colas gestionadas por dos gestores de colas diferentes. Los dos gestores de colas pueden estar ejecutándose en el mismo sistema z/OS, en sistemas z/OS diferentes o un gestor de colas puede estar en un sistema distribuido que ejecuta Advanced Message Security.

Los gestores de colas de ejemplo y las colas son:

```
BNK6        - Sending queue manager
BNK7        - Recipient queue manager
FIN.XFER.Q7 - Remote queue on BNK6
FIN.RCPT.Q7 - Local queue on BNK7
```

Nota: En este ejemplo, BNK6 y BNK7 son los gestores de colas que se ejecutan en diferentes sistemas z/OS con el mismo nombre.

Se utilizan estos usuarios:

```
WMQBNK6  - AMS task user on BNK6
WMQBNK7  - AMS task user on BNK7
TELLER5  - Sending user on BNK6
FINADM2  - Recipient user on BNK7
```

Los pasos para configurar este escenario son:

Crear el certificado de usuario

En este ejemplo, se necesitan dos certificados de usuario. Estos son el certificado del usuario emisor necesario para firmar mensajes y el certificado del usuario receptor necesario para cifrar y descifrar los datos de mensajes. El usuario emisor es 'TELLER5' y el usuario receptor es 'FINADM2'.

El certificado de la CA (Certificate Authority) también es necesario.El certificado de la CA es el certificado de la autoridad que ha emitido el certificado del usuario. Puede ser una cadena de certificados. Si es así, todos los certificados de la cadena serán necesarios en el conjunto de claves del usuario de la tarea Advanced Message Security, en este caso el usuario WMQBNK7.

Se puede crear un certificado de CA mediante el mandato RACDCERT de RACF. Este certificado se utiliza para emitir los certificados de usuario. Por ejemplo:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('BCOCA') O('BCO') C('US'))
KEYUSAGE(CERTSIGN) WITHLABEL('BCOCA')
```

Este mandato RACDCERT crea un certificado de CA que se puede utilizar para emitir un certificado de usuario para el usuario 'TELLER5' y el usuario 'FINADM2'. Por ejemplo:

```
RACDCERT ID(TELLER5) GENCERT SUBJECTSDN(CN('Teller5') O('BCO') C('US'))
WITHLABEL('Teller5') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(FINADM2) GENCERT SUBJECTSDN(CN('FinAdm2') O('BCO') C('US'))
WITHLABEL('FinAdm2') SIGNWITH(CERTAUTH LABEL('BCOCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)
```

La instalación tendrá procedimientos para seleccionar o crear un certificado de CA, así como procedimientos para emitir los certificados y distribuirlos en los sistemas relevantes.

Cuando se exportan e importan estos certificados, Advanced Message Security requieren:

- El certificado de CA (cadena).
- El certificado del usuario emisor y su clave privada.
- El certificado del usuario receptor y su clave privada.

Si está utilizando RACF, se puede utilizar el mandato RACDCERT EXPORT para exportar los certificados a un conjunto de datos y se puede utilizar el mandato RACDCERT ADD para importar certificados desde el conjunto de datos.

Para obtener más información sobre estos y otros mandatos RACDCERT, consulte RACDCERT (Gestionar certificados digitales RACF) en la publicación *z/OS: Security Server RACF Command Language Reference*.

En este caso, los certificados son necesarios en el sistema z/OS que ejecuta el gestor de colas BNK6 y BNK7.

En este ejemplo, los certificados de emisor y receptor se deben importar al sistema z/OS que ejecuta BNK6 y los certificados de CA se deben importar al sistema z/OS que ejecuta BNK7. Cuando se han importado los certificados, los certificados de usuario requieren el atributo TRUST. El mandato RACDCERT ALTER se puede utilizar para añadir el atributo TRUST al certificado. Por ejemplo:

En BNK6:

```
RACDCERT ID(TELLER5) ALTER (LABEL('Teller5')) TRUST
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

En BNK7:

```
RACDCERT ID(FINADM2) ALTER (LABEL('FinAdm2')) TRUST
```

Conectar los certificados a los conjuntos de claves relevantes

Una vez creados o importados los certificados necesarios, deben conectarse a los conjuntos de claves de usuario adecuados en los sistemas z/OS que ejecutan BNK6 y BNK7.

Para crear los conjuntos de claves utilice el mandato RACDCERT ADDRING:

En BNK6:

```
RACDCERT ID(WMQBNK6) ADDRING(drq.ams.keyring)
RACDCERT ID(TELLER5) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario de la tarea de Advanced Message Security y un conjunto de claves para el usuario emisor en BNK6. Tenga en cuenta que el nombre del conjunto de claves drq.ams.keyring es obligatorio y que el nombre distingue entre mayúsculas y minúsculas.

En BNK7:

```
RACDCERT ID(WMQBNK7) ADDRING(drq.ams.keyring)
RACDCERT ID(FINADM2) ADDRING(drq.ams.keyring)
```

Esto crea un conjunto de claves para el usuario de la tarea de Advanced Message Security y un conjunto de claves para el usuario receptor en BNK7.

Una vez creados los conjuntos de claves, se pueden conectar los certificados relevantes.

En BNK6:

```
RACDCERT ID(WMQBNK6) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) USAGE(SITE))

RACDCERT ID(TELLER5) CONNECT(ID(TELLER5) LABEL('Teller5')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

En BNK7:

```
RACDCERT ID(WMQBNK7) CONNECT(CERTAUTH LABEL('BCOCA')
RING(drq.ams.keyring))

RACDCERT ID(FINADM2) CONNECT(ID(FINADM2) LABEL('FinAdm2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

Los certificados del usuario emisor y el usuario receptor se deben conectar como DEFAULT. Si el usuario tiene más de un certificado en drq.ams.keyring, se utiliza el certificado predeterminado para fines de cifrado y descifrado.

En BNK6, el certificado del usuario emisor también debe estar conectado al conjunto de claves del usuario de la tarea Advanced Message Security con USAGE(SITE). Esto es debido a que la tarea Advanced Message Security necesita la clave pública del receptor cuando cifra los datos de mensajes. USAGE(SITE) impide que se pueda acceder a la clave privada desde el conjunto de claves.

Advanced Message Security no reconoce la creación y modificación de los certificados hasta que se ha detenido y reiniciado o hasta que se emita el mandato z/OS **MODIFY** para renovar la configuración de certificados de Advanced Message Security. Por ejemplo:

En BNK6:

```
F BNK6AMSM,REFRESH,KEYRING
```

En BNK7:

```
F BNK7AMSM,REFRESH,KEYRING
```

Crear las políticas de Advanced Message Security

En este ejemplo, los mensajes protegidos por privacidad los coloca en la cola FIN.XFER.Q7 en BNK6 una aplicación que se ejecuta como el usuario 'TELLER5' y los recupera de la cola local FIN.RCPT.Q7 en BNK7 una aplicación que se ejecuta como el usuario 'FINADM2' , por lo tanto, se requieren dos políticas de Advanced Message Security.

El programa de utilidad política de seguridad de mensaje (CSQ0UTIL) Advanced Message Security se crean utilizando el programa de utilidad CSQ0UTIL que se describe en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).

Utilice el programa de utilidad CSQ0UTIL para ejecutar el mandato siguiente para definir una política de privacidad para la cola remota en BNK6:

```
setmqspl -m BNK6 -p FIN.XFER.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK6. El nombre de política y la cola asociada es FIN.XFER.Q7. El algoritmo que se utiliza para generar la firma del emisor es SHA1, el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US' y el nombre del usuario receptor es 'CN=FinAdm2,O=BCO,C=US'. El algoritmo que se utiliza para cifrar datos de mensajes es 3DES.

Utilice también el programa de utilidad CSQ0UTIL para ejecutar el mandato siguiente para definir una política de privacidad para la cola local en BNK7:

```
setmqspl -m BNK7 -p FIN.RCPT.Q7 -s SHA1 -e 3DES -a CN=Teller5,O=BCO,C=US -r
CN=FinAdm2,O=BCO,C=US
```

En esta política, el gestor de colas se identifica como BNK7. El nombre de política y la cola asociada es FIN.RCPT.Q7. El algoritmo esperado para la firma del emisor es SHA1, el nombre distinguido (DN) del usuario emisor es 'CN=Teller5,O=BCO,C=US' y el nombre del usuario receptor es 'CN=FinAdm2,O=BCO,C=US'. El algoritmo que se utiliza para descifrar datos de mensajes es 3DES.

Después de definir dos políticas, reinicie los gestores de colas BNK6 y BNK7, o utilice el mandato z/OS **MODIFY** para renovar la configuración de política de Advanced Message Security. Por ejemplo:

En BNK6:

```
F BNK6AMSM,REFRESH,POLICY
```

En BNK7:

```
F BNK7AMSM,REFRESH,POLICY
```

Guía de inicio rápido para AMS con clientes Java

Utilice esta guía para aprender a configurar rápidamente Advanced Message Security para proporcionar seguridad de mensajes para las aplicaciones Java que se conectan utilizando enlaces de cliente. Cuando lo haya completado, habrá creado un almacén de claves para verificar las identidades de usuario y habrá definido políticas de firma/cifrado para su gestor de colas.

Antes de empezar

Asegúrese de que tiene instalados los componentes adecuados, tal como se describe en la **Guía de inicio rápido** (Windows o UNIX).

1. Crear un gestor de colas y una cola

Acerca de esta tarea

Todos los ejemplos siguientes utilizan una cola denominada TEST.Q para pasar mensajes entre aplicaciones. Advanced Message Security utiliza interceptores para firmar y cifrar mensajes en el momento en que los mensajes entran en la infraestructura de IBM MQ a través de la interfaz estándar de IBM MQ. La configuración básica se realiza en IBM MQ y se define en los pasos siguientes.

Procedimiento

1. Crear un gestor de colas

```
crtmqm QM_VERIFY_AMS
```

2. Inicie el gestor de colas

```
strmqm QM_VERIFY_AMS
```

3. Cree e inicie un escucha especificando los mandatos siguientes en **runmqsc** para el gestor de colas QM_VERIFY_AMS

```
DEFINE LISTENER(AMS.LSTR) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR)
START LISTENER(AMS.LSTR)
```

4. Cree un canal para que se conecten las aplicaciones especificando el siguiente mandato en **runmqsc** para el gestor de colas QM_VERIFY_AMS

```
DEFINE CHANNEL(AMS.SVRCONN) CHLTYPE(SVRCONN)
```

5. Cree una cola denominada TEST.Q especificando el siguiente mandato en **runmqsc** para el gestor de colas QM_VERIFY_AMS

```
DEFINE QLOCAL(TEST.Q)
```

Resultados

Si el procedimiento se ha ejecutado correctamente, el siguiente mandato especificado en **runmqsc** muestra detalles sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Crear y autorizar usuarios

Acerca de esta tarea

En este caso de ejemplo existen dos usuarios: alice, el emisor, y bob, el receptor. Para utilizar la cola de aplicación, estos usuarios deben tener autorización para utilizarla. Asimismo, para utilizar correctamente las políticas de protección definidas en este caso de ejemplo, estos usuarios deben tener acceso a algunas colas del sistema. Para obtener más información sobre el mandato **setmqaut** , consulte **setmqaut**.

Procedimiento

1. Cree los dos usuarios, tal como se describe en la **Guía de inicio rápido** (Windows o UNIX) correspondiente a su plataforma.
2. Autorice a los usuarios para conectarse con el gestor de colas y para trabajar con la cola

```
setmqaut -m QM_VERIFY_AMS -t qmgr -p alice -p bob +connect +inq
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p alice +put
setmqaut -m QM_VERIFY_AMS -n TEST.Q -t queue -p bob +get +inq +browse
```

3. También debe permitir que los dos usuarios examinen la cola de políticas del sistema y coloquen mensajes en la cola de errores.

```
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
```

Atención: IBM MQ optimiza el rendimiento almacenando en memoria caché las políticas para que no tenga que examinar los registros para obtener detalles de política en el SYSTEM.PROTECTION.POLICY.QUEUE en todos los casos.

IBM MQ no almacena en memoria caché todas las políticas disponibles. Si hay un número alto de políticas, IBM MQ almacena en memoria caché un número limitado de políticas. Por lo tanto, si el gestor de colas tiene un número bajo de políticas definidas, no es necesario proporcionar la opción de examinar a SYSTEM.PROTECTION.POLICY.QUEUE.

Sin embargo, debe otorgar autorización para examinar esta cola, en caso de que haya un gran número de políticas definidas, o si está utilizando clientes antiguos. El sistema SYSTEM.PROTECTION.ERROR.QUEUE se utiliza para colocar mensajes de error generados por el código AMS. La autorización de colocación sobre esta cola sólo se comprueba cuando se intenta colocar un mensaje de error en la cola. La autorización de colocación sobre la cola no se comprueba cuando intenta transferir u obtener un mensaje de una cola protegida por AMS.

Resultados

Se crean los usuarios y se les otorgan las autorizaciones necesarias.

Qué hacer a continuación

Para verificar si los pasos se han realizado correctamente, utilice los ejemplos `JmsProducer` y `JmsConsumer`, tal como se describe en la sección “7. Probar la configuración” en la página 608.

3. Crear la base de datos de claves y certificados

Acerca de esta tarea

Para cifrar el mensaje para el los interceptores es necesario la clave pública de los usuarios emisores. Por lo tanto, se deben crear la base de datos de claves de identidades de usuario que están correlacionadas con claves públicas y privadas. En el sistema real, donde los usuarios y las aplicaciones están distribuidos en varios sistemas, cada usuario tendría su propio almacén de claves privado. De forma similar, en esta guía, creamos bases de datos de claves para `alice` y bob y compartimos los certificados de usuario entre ellos.

Nota: En esta guía, utilizamos aplicaciones de ejemplo escritas en Java que se conectan mediante enlaces de cliente. Si tiene previsto utilizar aplicaciones Java utilizando enlaces locales o las aplicaciones C, debe crear un almacén de claves CMS y los certificados mediante el mandato **runmqakm**. Esto se muestra en la **Guía de inicio rápido** (Windows o UNIX).

Procedimiento

1. Cree un directorio en el que crear el almacén de claves, por ejemplo `/home/alice/.mqs`. Si lo desea, puede crearlo en el mismo directorio que ha utilizado la **Guía de inicio rápido** (Windows o UNIX) para su plataforma.

   Nota: Este directorio se conoce como *keystore-dir* en los siguientes pasos
2. Cree un nuevo almacén de claves y un certificado que identifique al usuario `alice` para utilizarlo en el cifrado

Nota: El mandato **keytool** es parte del 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:

- Si *keystore-dir* contiene espacios, debe escribir el nombre completo del almacén de claves entre comillas
- Se recomienda utilizar una contraseña fuerte para proteger el almacén de claves.
- A los efectos de esta guía, utilizamos un certificado autofirmado que se puede crear sin utilizar una entidad emisora de certificados. Para los sistemas de producción, se recomienda no utilizar certificados autofirmados, sino certificados firmados por una entidad emisora de certificados.
- El parámetro **alias** especifica el nombre para el certificado, que los interceptores buscarán para recibir la información necesaria.
- El parámetro **dname** especifica los detalles del **Nombre distinguido** (DN), que debe ser exclusivo para cada usuario.

3. En UNIX, asegúrese de que el almacén de claves sea legible

```
chmod +r keystore-dir/keystore.jks
```

4. Repita los pasos del 1 al 4 para el usuario bob

Resultados

Los dos usuarios `alice` y bob tienen cada uno un certificado autofirmado.

4. Crear keystore.conf

Acerca de esta tarea

Debe apuntar los interceptores de Advanced Message Security al directorio donde residen las bases de datos y certificados. Esto se realiza mediante el archivo `keystore.conf`, que contiene esa información como texto sin formato. Cada usuario debe tener un archivo `keystore.conf` separado. Este paso debe realizarse para `alice` y bob.

Ejemplo

Para este caso de ejemplo, el contenido de `keystore.conf` para `alice` es como el siguiente:

```
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 este caso de ejemplo, el contenido de `keystore.conf` para bob es como el siguiente:

```
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:

- La vía de acceso del archivo de almacén de claves se debe especificar sin ninguna extensión de archivo.

- Si ya tiene un archivo `keystore.conf` porque ha seguido las instrucciones de la Guía de inicio rápido (Windows o UNIX), puede editar el archivo existente para añadir estas líneas.
- Para obtener más información, consulte "Estructura del archivo de configuración del almacén de claves (keystore.conf) para AMS" en la página 617.

5. Compartir certificados

Acerca de esta tarea

Comparta los certificados entre los dos almacenes de claves para que cada usuario pueda identificar correctamente al otro. Esto se realiza extrayendo el certificado de cada usuario e importándolo en el almacén de claves de otro usuario.

Nota: Los términos *extraer* y *exportar* se utilizan de forma diferente por parte de distintas herramientas de certificado. Por ejemplo, la herramienta de mandato IBM GSKit **strmqikm** (ikeyman) hace una distinción entre *extraer* certificados (claves públicas) y *exportar* claves privadas. Esta distinción es extremadamente importante para las herramientas que ofrecen ambas opciones, ya que utilizar *exportar* por error comprometería por completo la aplicación pasando su clave privada. Puesto que la distinción es tan importante, la documentación de IBM MQ se esfuerza en utilizar estos términos de forma coherente. Sin embargo, la herramienta keytool Java proporciona una opción de línea de mandatos llamada *exportcert* que solo extrae la clave pública. Por estos motivos, el procedimiento siguiente hace referencia a *extraer* certificados utilizando la opción *exportcert*.

Procedimiento

1. Extraiga el certificado que identifica a `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 el certificado que identifica a `alice` al almacén de claves que utilizará bob. Cuando se le solicite, indique que no confía en este 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 los pasos para bob

Resultados

Los dos usuarios `alice` y bob pueden identificar ahora correctamente al otro mediante la creación y compartición de certificados autofirmados.

Qué hacer a continuación

Verifique que un certificado esté en el almacén de claves ejecutando los siguientes mandatos que imprimen los detalles:

```
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. Definir la política de cola

Acerca de esta tarea

Después de crear el gestor de colas y preparar interceptores para interceptar mensajes y acceder a claves de cifrado, podemos comenzar a definir políticas de protección en `QM_VERIFY_AMS` mediante el mandato `setmqspl`. Consulte setmqspl para obtener más información sobre este mandato. Cada nombre de política debe ser el mismo que el nombre de cola al que se debe aplicar.

Ejemplo

Esto es un ejemplo de una política definida para la cola TEST.Q, firmada por el usuario `alice` mediante el algoritmo SHA1 y cifrada utilizando el algoritmo AES de 256 bits para el usuario 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: Los DN coinciden exactamente con los especificados en el certificado del usuario respectivo de la base de datos de claves.

Qué hacer a continuación

Para verificar la política que ha definido, emita el mandato siguiente:

```
dspmqspl -m QM_VERIFY_AMS
```

Para mostrar los detalles de la política como un conjunto de mandatos `setmqspl`, utilice el distintivo `-export`. Esto permite almacenar políticas ya definidas:

```
dspmqspl -m QM_VERIFY_AMS -export >restore_my_policies.bat
```

7. Probar la configuración

Antes de empezar

Asegúrese de que la versión de Java que está utilizando ya tiene instalados los archivos de políticas JCE sin restricciones.

Nota: La versión de Java proporcionada en la instalación de IBM MQ ya tiene estos archivos de política. Puede encontrarse en *MQ_INSTALLATION_PATH*`/java/bin`.

Acerca de esta tarea

Puede ejecutar programas diferentes bajo usuarios diferentes para verificar si la aplicación se ha configurado debidamente. Consulte la **Guía de inicio rápido** (Windows o UNIX) correspondiente su plataforma para obtener detalles sobre cómo ejecutar programas con usuarios distintos.

Procedimiento

1. Para ejecutar estas aplicaciones de ejemplo JMS, utilice el valor de CLASSPATH correspondiente a su plataforma, tal como se muestra en Variables de entorno utilizadas por las IBM MQ classes for JMS para asegurarse de que se incluye el directorio de ejemplos.
2. Como usuario `alice`, coloque un mensaje utilizando una aplicación de ejemplo, conectarse como un cliente:

   ```
   java JmsProducer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

3. Como usuario bob, obtenga un mensaje utilizando una aplicación de ejemplo, conectarse como un cliente:

   ```
   java JmsConsumer -m QM_VERIFY_AMS -d TEST.Q -h localhost -p 1414 -l AMS.SVRCONN
   ```

Resultados

Si la aplicación se ha configurado debidamente para ambos usuarios, el mensaje del usuario `alice` se visualiza cuando bob ejecuta la aplicación de obtención.

Protección de colas remotas

Para proteger completamente las colas remotas, deben establecerse políticas en la cola remota y en la cola local a las que se transmiten los mensajes.

Cuando se pone un mensaje en una cola remota, Advanced Message Security intercepta la operación y procesa el mensaje según un conjunto de políticas definido para la cola remota. Por ejemplo, para una política de cifrado, el mensaje se cifra antes de que se pase a IBM MQ para su proceso. Después de procesar el mensaje, Advanced Message Security lo coloca en una cola remota, IBM MQ coloca el mensaje en la cola de transmisión asociada y lo reenvía al gestor de colas de destino y cola de destino.

Cuando la operación GET se realiza en la cola local, Advanced Message Security intenta descifrar el mensaje de acuerdo con la política definida en la cola local. Para que la operación sea satisfactoria, la política utilizada para descifrar el mensaje debe ser la misma que la utilizada para cifrarlo. Cualquier discrepancia provocará que se rechace el mensaje.

Si por cualquier motivo las políticas no se puede definir al mismo tiempo, se proporciona un mecanismo de despliegue gradual. La política se puede definir en una cola local con el distintivo de tolerancia activado, el cual indica que se pase por alto la política asociada a una cola cuando el intento de recuperar un mensaje de la cola afecte a un mensaje que no tenga definida la política de seguridad. En este caso, GET intentará descifrar el mensaje, pero permitirá la entrega de los mensaje no cifrados. De esta forma se pueden definir políticas en colas remotas después de proteger (y probar) las colas locales.

Recuerde: Elimine el distintivo de tolerancia una vez que concluya el despliegue de Advanced Message Security.

Referencia relacionada

setmqspl (establecer política de seguridad)

Direccionamiento de mensajes protegidos mediante IBM Integration Bus

Advanced Message Security puede proteger los mensajes en una infraestructura donde está instalado IBM Integration Buso WebSphere Message Broker 8.0.0.1 (o posterior). Debe comprender la naturaleza de ambos productos antes de aplicar la seguridad en el entorno de IBM Integration Bus.

Acerca de esta tarea

Advanced Message Security proporciona seguridad global para la carga útil del mensaje. Esto significa que sólo los interlocutores especificados como remitentes y destinatarios válidos de un mensaje pueden emitirlo o recibirlo. Esto significa que para proteger los mensajes que fluyen por IBM Integration Bus, puede permitir que IBM Integration Bus procese los mensajes sin conocer su contenido (Caso de ejemplo 1) o convertirlo en un usuario con autorización para recibir y enviar mensajes (Caso de ejemplo 2).

Caso de ejemplo 1 - El bus de integración no puede ver el contenido del mensaje

Antes de empezar

Debe tener su IBM Integration Bus conectado a un gestor de colas existente. Sustituya *QMgrName* por este nombre de gestor de colas existente en los mandatos siguiente.

Acerca de esta tarea

En este caso de ejemplo, Alice coloca un mensaje protegido en una cola de entrada `QIN`. Basándose en la propiedad de mensaje `routeTo`, el mensaje se direcciona a *bob's* (`QBOB`),[1](`QCECIL`), o la cola predeterminada (`QDEF`). El direccionamiento es posible porque Advanced Message Security sólo protege la carga útil del mensaje y no sus cabeceras y propiedades, que permanecen desprotegidos y pueden ser leídos por IBM Integration Bus. Advanced Message Security es utilizado sólo por *alice*, *bob* y *cecil*. No es necesario instalarlo ni configurarlo para IBM Integration Bus.

IBM Integration Bus recibe el mensaje protegido desde la cola del alias no protegida para evitar cualquier intento de descifrar el mensaje. Si desea utilizar la cola protegida directamente, el mensaje debe colocarse en la cola de mensajes no entregados como imposible de descifrar. IBM Integration Bus

[1] cecil's

direcciona el mensaje, que llega a la cola de destino sin modificar. Por lo tanto, es todavía firmado por el autor original (tanto *bob* como *cecil* sólo aceptarán mensajes enviados por *alice*) y está protegido como antes (sólo *bob* y *cecil* pueden leerlo). IBM Integration Bus coloca el mensaje direccionado en un alias no protegido. Los destinatarios recuperan el mensaje de una cola de salida protegida donde AMS descifrará de forma transparente el mensaje.

Procedimiento

1. Configure *alice*, *bob* y *cecil* para que utilicen Advanced Message Security, tal como se describe en la **Guía de inicio rápido** (Windows o UNIX).

   Asegúrese de que los pasos siguientes se hayan completado:

   - Crear y autorizar usuarios
   - Crear la base de datos de claves y certificados
   - Crear keystore.conf
2. Proporcione el certificado de *alice* a *bob* y *cecil*, de modo que *alice* pueda ser identificada por ellos durante la comprobación de firmas digitales en los mensajes.

   Hágalo extrayendo el certificado que identifica a *alice* a un archivo externo y, después, añadiendo el certificado extraído a los almacenes de claves *de bob* y *de cecil*. Es importante que utilice el método descrito en la tarea 5 de **. Compartir certificados** en la **Guía de inicio rápido** (Windows o UNIX).
3. Proporcione los certificados de *bob* y *cecil* a *alice*, con lo que *alice* podrá enviar mensajes cifrados a *bob* y *cecil*.

   Hágalo utilizando el método especificado en el paso anterior.
4. En el gestor de colas, defina las colas locales denominadas `QIN`, `QBOB`, `QCECIL` y `QDEF`.

   ```
   DEFINE QLOCAL(QIN)
   ```
5. Configure la política de seguridad de la cola `QIN` para una configuración elegible. Utilice la misma configuración para las colas `QBOB`, `QCECIL` y `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 caso de ejemplo presupone la política de seguridad en la que *alice* es el único emisor autorizado y *bob* y *cecil* son los destinatarios.
6. Defina las colas de alias `AIN`, `ABOB` y `ACECIL` que hacen referencia a las colas locales `QIN`, `QBOB` y `QCECIL`, respectivamente.

   ```
   DEFINE QALIAS(AIN) TARGET(QIN)
   ```
7. Verifique que la configuración de seguridad para los alias especificados en el paso anterior no está presente; de lo contrario, establezca su política en NONE.

   ```
   dspmqspl -m QMgrName -p AIN
   ```
8. En IBM Integration Bus, cree un flujo de mensajes para direccionar los mensajes que llegan a la cola de alias `AIN` al nodo BOB, CECIL o DEF, dependiendo de la propiedad `routeTo` del mensaje. Para ello:
   a) Cree un nodo `MQInput` denominado `IN` y asigne el alias `AIN` como su nombre de cola.
   b) Cree nodos `MQOutput` denominados `BOB`, `CECIL` y `DEF` y asigne las colas de alias `ABOB`, `ACECIL` y `ADEF` como sus nombres de colas respectivos.
   c) Cree un nodo de ruta y asígnele el nombre TEST.
   d) Conecte el nodo `IN` al terminal de entrada del nodo TEST.

e) Cree los terminales de salida bob y `cecil` para el nodo TEST.

f) Conecte el terminal de salida bob al nodo BOB.

g) Conecte el terminal de salida `cecil` al nodo `CECIL`.

h) Conecte el nodo DEF al terminal de salida predeterminado.

i) Aplique las reglas siguientes:

```
$Root/MQRFH2/usr/routeTo/text()="bob"
$Root/MQRFH2/usr/routeTo/text()="cecil"
```

9. Despliegue el flujo de mensajes en el componente de ejecución de IBM Integration Bus.
10. Ejecutándose como el usuario `Alice`, coloque un mensaje que también contenga una propiedad de mensaje denominada `routeTo` con un valor bob o `cecil`. Para ello, ejecute la aplicación de ejemplo **amqsstm**.

```
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. Ejecutándose como el usuario *bob*, recupere el mensaje de la cola QBOB utilizando la aplicación de ejemplo **amqsget**.

Resultados

Cuando *alice* coloca un mensaje en la cola `QIN`, el mensaje queda protegido. IBM Integration Bus recupera el mensaje con el formato protegido del alias `AIN`. IBM Integration Bus decide adónde direccionar el mensaje examinando la propiedad `routeTo`, la cual no está cifrada, como ocurre con todas las propiedades. IBM Integration Bus coloca el mensaje en el alias no protegido apropiado para evitar su protección ulterior. Cuando *bob* o *cecil* reciben el mensaje de la cola, se descifra el mensaje y se verifica la firma digital.

Caso de ejemplo 2 - El bus de integración puede ver el contenido del mensaje

Acerca de esta tarea

En este ejemplo, un grupo de usuarios está autorizado para enviar mensajes a IBM Integration Bus. Otro grupo está autorizado para recibir los mensajes que ha creado IBM Integration Bus. La transmisión entre las partes y IBM Integration Bus no puede ser interceptada.

Tenga en cuenta que IBM Integration Bus lee las políticas de protección y los certificados una sola vez, por lo que debe volver a cargar el grupo de ejecución después de realizar cualquier actualización en las políticas de protección para que los cambios surtan efecto.

```
mqsireload execution-group-name
```

Si se considera que IBM Integration Bus es un interlocutor autorizado con permiso para leer o firmar la carga útil del mensaje, debe configurar Advanced Message Security para el usuario encargado de iniciar el servicio de IBM Integration Bus. Tenga en cuenta que no es necesariamente el mismo usuario que realiza operaciones PUT o GET para mensajes de las colas ni el usuario que crea y despliega aplicaciones de IBM Integration Bus.

Procedimiento

1. Configure *alice*, *bob*, *cecil* y *dave* y el usuario de servicio de IBM Integration Bus para que se utilice Advanced Message Security como se describe en la **Guía de inicio rápido** (Windows o UNIX).

   Asegúrese de que los pasos siguientes se hayan completado:

   - Crear y autorizar usuarios
   - Crear la base de datos de claves y certificados
   - Crear keystore.conf
2. Proporcione los certificados de *alice*, *bob*, *cecil* y *dave* al usuario de servicio IBM Integration Bus.

   Hágalo extrayendo cada uno de los certificados que identifica a *alice*, *bob*, *cecil* y *dave* en archivos externos y, después, añadiendo los certificados extraídos al almacén de claves de IBM Integration Bus. Es importante que utilice el método descrito en la tarea 5 de **. Compartir certificados** en la **Guía de inicio rápido** (Windows o UNIX).
3. Proporcione el certificado del usuario de servicio de IBM Integration Bus a *alice*, *bob*, *cecil* y *dave*.

   Hágalo utilizando el método especificado en el paso anterior.

   Nota: *Alice* y *Bob* necesitan el certificado del usuario de servicio de IBM Integration Bus para cifrar los mensajes correctamente. El usuario de servicio de IBM Integration Bus necesita los certificados de *Alice* y *Bob* para verificar los autores de los mensajes. El usuario de servicio de IBM Integration Bus necesita los certificados de *Cecil* y *Dave* para cifrar los mensajes destinados a ellos. *cecil* y *dave* necesitan el certificado del usuario de servicio de IBM Integration Bus para verificar si el mensaje procede de IBM Integration Bus.
4. Defina una cola local denominada `IN` y defina la política de seguridad con *alice* y *bob* especificados como autores y el usuario de servicio para IBM Integration Bus especificado como destinatario:

   ```
   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 una cola local llamada OUT y defina la política de seguridad con el usuario de servicio para IBM Integration Bus especificado como autor, y *cecil* y *dave* especificados como destinatarios:

   ```
   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. En IBM Integration Bus, cree un flujo de mensajes con un nodo `MQInput` y `MQOutput`. Configure el nodo `MQInput` para utilizar la cola `IN` y el nodo `MQOutput` para utilizar la cola OUT.
7. Despliegue el flujo de mensajes en el componente de ejecución de IBM Integration Bus.
8. Ejecutándose como el usuario *Alice* o *Bob*, coloque un mensaje en la cola `IN` utilizando la aplicación de ejemplo **amqsput**.
9. Ejecutándose como el usuario *Cecil* o *Dave*, recupere el mensaje de la cola OUT utilizando la aplicación de ejemplo **amqsget**.

Resultados

Los mensajes enviados por *Alice* o *Bob* a la cola de entrada `IN` están cifrados, por lo que sólo puede leerlos IBM Integration Bus. IBM Integration Bus solo acepta mensajes de *alice* y *bob* y rechaza los otros. Los mensajes aceptados se procesarán debidamente y, a continuación, se firmarán y cifrarán con las claves de *Cecil* y *Dave* antes de colocarse en la cola de salida OUT. Sólo lo pueden leer *Cecil* y *Dave* y se rechazarán los mensajes no firmados por IBM Integration Bus.

Utilización de Advanced Message Security con Managed File Transfer

Este caso de ejemplo explica cómo configurar Advanced Message Security para proporcionar privacidad de mensajes para los datos que se envían a través de Managed File Transfer.

Antes de empezar

Compruebe que tiene el componente Advanced Message Security instalado en la instalación de IBM MQ que aloje las colas utilizadas por Managed File Transfer que quiera proteger.

Si los agentes de Managed File Transfer se conectan en modalidad de enlaces, asegúrese de tener instalado también el componente GSKit en su instalación local.

Acerca de esta tarea

Cuando se interrumpe la transferencia de datos entre dos agentes de Managed File Transfer, es probable que queden datos confidenciales desprotegidos en las colas de IBM MQ subyacentes que se utilizan para gestionar la transferencia. En este caso de ejemplo, aprenderemos a configurar y utilizar Advanced Message Security para proteger estos datos en las colas de Managed File Transfer.

En este escenario, consideramos una topología simple que consta de una máquina con dos colas Managed File Transfer y dos agentes, AGENT1 y AGENT2, que comparten un único gestor de colas, tal como se describe en el escenario Visión general del escenario. Ambos agentes se conectan de la misma forma, ya sea en la modalidad de enlaces o en la modalidad de cliente.

1. Creación de certificados

Antes de empezar

Este escenario utiliza un modelo simple donde un usuario `ftagent` de un grupo FTAGENTS se utiliza para ejecutar los procesos Managed File Transfer Agent. Si utiliza su propios nombres de usuario y grupo, cambie los mandatos según corresponda.

Acerca de esta tarea

Advanced Message Security utiliza criptografía de clave pública para firmar o cifrar mensajes en colas protegidas.

Nota:

- Si los agentes de Managed File Transfer se ejecutan en modalidad de enlaces, los mandatos que utiliza para crear un almacén de claves de CMS (Cryptographic Message Syntax) se detallan en la **Guía de inicio rápido** (Windows o UNIX) correspondiente a su plataforma.
- Si los agentes de Managed File Transfer se ejecutan en modalidad de cliente, los mandatos que necesitará para crear un JKS (almacén de claves Java) se detallan en “Guía de inicio rápido para AMS con clientes Java” en la página 603.

Procedimiento

1. Cree un certificado autofirmado para identificar al usuario `ftagent` como se describe en la Guía de inicio rápido correspondiente.
   Utilice un nombre distinguido (DN) de la siguiente manera:

   ```
   CN=ftagent, OU=MFT, O=<organisation>, L=<location>, ST=<state>, C=<country>
   ```

2. Cree un archivo `keystore.conf` para identificar la ubicación del almacén de claves y el certificado que contiene como se describe en la Guía de inicio rápido correspondiente.

2. Configuración de protección de mensajes

Acerca de esta tarea

Debe definir una política de seguridad para la cola de datos que utiliza AGENT2, mediante el mandato **setmqspl**. En este caso de ejemplo, se utiliza el mismo usuario para iniciar ambos agentes y, por lo tanto, el DN firmante y receptor son iguales y coinciden con el certificado que hemos generado.

Procedimiento

1. Concluya los agentes de Managed File Transfer para preparar la protección mediante el mandato **fteStopAgent**.
2. Cree una política de seguridad para proteger la cola SYSTEM.FTE.DATA.AGENT2.

```
setmqspl -m hubQM -p SYSTEM.FTE.DATA.AGENT2 -s SHA1 -a "CN=ftagent, OU=MFT,
O=<organisation>, L=<location>, ST=<state>, C=<country>"
-e AES128 -r "CN=ftagent, OU=MFT, O=<organisation>, L=<location>, ST=<state>, C=<country>"
```

3. Asegúrese de que el usuario que ejecuta el proceso de Managed File Transfer Agent tiene acceso para examinar la cola de políticas del sistema y colocar mensajes en la cola de errores.

```
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 los agentes de Managed File Transfer mediante el mandato **fteStartAgent**.
5. Confirme que los agentes se hayan reiniciado satisfactoriamente mediante el mandato **fteListAgents** y verifique que los agentes tengan el estado READY.

Resultados

Ahora puede enviar transferencias desde AGENT1 a AGENT2, y el contenido del archivo se transmitirá de forma segura entre los dos agentes.

Descripción general de la instalación de Advanced Message Security

Instalar el componente Advanced Message Security en varias plataformas.

Acerca de esta tarea

Para obtener información sobre los procedimientos de instalación, consulte Instalación de Advanced Message Security en multiplataformas e Instalación de Advanced Message Security en z/OS.

Tareas relacionadas

Desinstalación de Advanced Message Security

z/OS Auditoría en z/OS

Advanced Message Security (AMS) para z/OS proporciona medios para una auditoría opcional de las operaciones mediante aplicaciones en colas protegidas con políticas. Cuando se habilita, se generan registros de auditoría de IBM System Management Facility (SMF) acerca de si estas operaciones en colas protegidas mediante seguridad se han realizado o no correctamente. Las operaciones auditadas son MQPUT, MQPUT1 y MQGET.

La auditoría está inhabilitada de forma predeterminada, sin embargo, puede activar la auditoría configurando _AMS_SMF_TYPE y _AMS_SMF_AUDIT en el archivo _CEE_ENVFILE de Language Environment® configurado para el espacio de direcciones de AMS. Si desea más información, consulte Crear procedimientos para Advanced Message Security. Se utiliza _AMS_SMF_TYPE la variable para designar el tipo de registro SMF que es un número entre 128 y 255. Lo habitual es un tipo de registro SMF de 180, aunque no es obligatorio. La auditoría se inhabilita especificando un valor de 0. La variable _AMS_SMF_AUDIT configura si se crean registros de auditoría para las operaciones que son satisfactorias, las operaciones que fallan o ambas. Las opciones de auditoría también se pueden modificar dinámicamente mientras AMS está activo utilizando mandatos de operador. Para más información, consulte Trabajar con Advanced Message Security.

El registro SMF se define utilizando subtipos, siendo el subtipo 1 un suceso de auditoría general. El registro SMF contiene todos los datos relevantes para la solicitud que se está procesando.

El registro SMF se correlaciona con la macro CSQ0KSMF, la cual se proporciona en la biblioteca de destino SCSQMACS (tenga en cuenta el cero en el nombre de la macro). Si está escribiendo programas

de reducción de datos para los datos SMF, puede incluir esta macro de correlación como ayuda para el desarrollo y la personalización de las rutinas de post-proceso de SMF.

En los registros SMF generados por Advanced Message Security para z/OS, los datos se organizan en secciones. El registro consta de:

- una cabecera SMF estándar
- Una extensión de cabecera definida por Advanced Message Security para z/OS
- una sección del producto
- una sección de datos

La sección del producto del registro SMF siempre está presente en los registros generados por Advanced Message Security para z/OS. La sección de datos varía en función del subtipo. Actualmente, se define un subtipo y, por lo tanto, se utiliza una sola sección de datos.

SMF se describe en la publicación z/OS System Management Facilities (SA22-7630). Los tipos de registros válidos se describen en el miembro SMFPRMxx de sus conjunto de datos PARMLIB del sistema. Consulte la documentación de SMF para obtener más información.

Generador de informes de auditoría de Advanced Message Security (CSQ0USMF)

Advanced Message Security para z/OS proporciona una herramienta de generador de informes de auditoría llamada CSQ0USMF que se proporciona en la biblioteca SCSQAUTH de instalación. El JCL de ejemplo para ejecutar el programa de utilidadCSQ0USMF denominado CSQ40RSM se proporciona en la biblioteca de instalación SCSQPROC.

Antes de ejecutar el programa de utilidad CSQ0USMF, los registros SMF de tipo 180 se deben volcar desde los conjuntos de datos SMF del sistema a un conjunto de datos secuencial. Como ejemplo, este JCL vuelca los registros SMF del tipo 180 desde un conjunto de datos SMF y los transfiere a un conjunto de datos de destino:

```
//IFAUDUMP EXEC PGM=IFASMFDP
//INDD1 DD DSN=SYSn.MANn.syst,DISP=SHR
//OUTDD1 DD DSN=your.target.dataset,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
INDD(INDD1,OPTIONS(DUMP))
OUTDD(OUTDD1,TYPE(180))
/*
```

Debe verificar los nombres del conjunto de datos SMF que utiliza su instalación. El conjunto de datos de destino para los registros volcados debe tener un formato de registro de VBS y una longitud de registro de 32760.

Nota: Si se están utilizando corrientes de registro SMF, debe utilizar el programa IFASMFDL para volcar una corriente de registro en un conjunto de datos secuencial. Consulte Proceso de registros SMF de tipo 116 para ver un ejemplo del JCL utilizado.

El conjunto de datos de destino se puede utilizar como entrada para el programa de utilidad CSQ0USMF para generar un informe de auditoría AMS. Por ejemplo:

```
//STEP1 EXEC PGM=CSQ0USMF,
// PARM=('/ -SMFTYPE 180 -M qmgr')
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//    DD DSN=thlqual.SCSQAUTH,DISP=SHR
//SMFIN DD DSN=your.target.dataset,DISP=SHR
//
```

El programa CSQ0USMF acepta dos parámetros opcionales, que se listan en Tabla 97 en la página 616:

Tabla 97. Parámetros opcionales de CSQ0USMF		
Parámetro	**Valor**	**Descripción**
SMFTYPE	nnn	El tipo de registro SMF aplicable al informe de auditoría. El programa CSQ0USMF utiliza únicamente registros SMF que coinciden con el valor SMFTYPE cuando genera el informe. Si no especifica SMFTYPE, se utiliza un valor predeterminado de 180.
M	qmgr	El nombre del gestor de colas IBM MQ aplicable al informe de auditoría. Si no especifica el parámetro -M, el informe de auditoría incluirá todos los registros de auditoría para todos los gestores de colas representados en el conjunto de datos SMFIN.

Utilización de almacenes de claves y certificados

Para proporcionar protección de cifrado transparente para las aplicaciones de IBM MQ, Advanced Message Security utiliza el archivo de almacén de claves, donde se almacenan certificados de clave pública y una clave privada. En z/OS, se utiliza un conjunto de claves SAF en lugar del archivo del almacén de claves.

En Advanced Message Security, los usuarios y las aplicaciones se representan mediante identidades de la infraestructura de claves públicas (PKI). Este tipo de identidad se utiliza para firmar y cifrar mensajes. La identidad PKI está representada por el campo **Nombre distinguido (DN)** del asunto en un certificado asociado con mensajes firmados o cifrados. Un usuario o una aplicación que desee cifrar sus mensajes debe tener acceso al archivo de almacén de claves donde se almacenan los certificados y las claves privadas y públicas asociadas.

En Windows y UNIX la ubicación del almacén de claves se proporciona en el archivo de configuración del almacén de claves que, de forma predeterminada, es `keystore.conf`. Cada usuario de Advanced Message Security debe tener el archivo de configuración del almacén de claves que apunte a un archivo de almacén de claves. Advanced Message Security acepta los siguientes formatos de archivos de almacén de claves: `.kdb`, `.jceks`, `.jks`.

La ubicación predeterminada del archivo `keystore.conf` es:

- IBM i UNIX En UNIX y IBM i: `$HOME/.mqs/keystore.conf`
- Windows en Windows: *%HOMEDRIVE%%HOMEPATH%*`\.mqs\keystore.conf`

  Nota: La vía de acceso en Windows puede y debe especificar la letra de unidad si hay más de una letra de unidad disponible.

Si está utilizando un nombre de archivo y una ubicación del almacén de claves especificados, debe utilizar los mandatos siguientes

- Para Java: `java -D`*MQS_KEYSTORE_CONF=path/filename app_name*
- Para el cliente y servidor de C:
  - en UNIX and Linux: `export MQS_KEYSTORE_CONF=`*path/filename*
  - en Windows: `set MQS_KEYSTORE_CONF=`*path\filename*

Conceptos relacionados

“Nombres distinguidos de emisor en AMS” en la página 644
Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola. Un remitente utiliza su certificado para firmar un mensaje, antes de colocar el mensaje en una cola.

“Nombres distinguidos de destinatario en AMS” en la página 645
Los nombres distinguidos de destinatario identifican a usuarios que están autorizados a recuperar mensajes de una cola.

Estructura del archivo de configuración del almacén de claves (keystore.conf) para AMS

El archivo de configuración del almacén de claves (`keystore.conf`) apunta Advanced Message Security a la ubicación del almacén de claves adecuado.

Cada uno de los siguientes tipos de archivo de configuración tiene un prefijo:

CMS
Certificate Management System, las entradas de configuración tienen el prefijo: `cms.`

PKCS#11
Public Key Cryptography Standard #11, las entradas de configuración tienen el prefijo: `pkcs11.`

IBM i **PEM**
Formato Privacy Enhanced Mail, las entradas de configuración tienen el prefijo: `pem.`

JKS
Java KeyStore, las entradas de configuración tienen el prefijo: `jks.`

JCEKS
Java Cryptographic Encryption KeyStore, las entradas de configuración tienen el prefijo: `jceks.`

z/OS V 9.1.0 MQ Adv. VUE **JCERACFKS**
Java Cryptographic Encryption RACF keyring KeyStore, las entradas de configuración tienen el prefijo: `jceracfks`.

Importante: A partir de IBM MQ 9.0, no se tienen en cuenta los valores de `JCEKS.provider` y `JKS.provider`. Se utiliza el proveedor Bouncy Castle, junto con el suministro JCE/JCE que proporcione el JRE en uso. Para obtener más información, consulte “Soporte para JRE que no son deIBM con AMS” en la página 620.

Estructuras de ejemplo para almacenes de claves:

CMS

```
cms.keystore = /dir/keystore_file
cms.certificate = certificate_label
```

PKCS#11

```
pkcs11.library = dir\cryptoki.dll
pkcs11.certificate = certificatelabel
pkcs11.token = tokenlabel
pkcs11.token_pin = tokenpin
pkcs11.secondary_keystore = dir\signers
```

IBM i PEM

```
pem.private = /dir/keystore_file_private_key
pem.public = /dir/keystore_file_public_keys
pem.password = password
```

Java JKS

```
jks.keystore = dir/Keystore
jks.certificate = certificate_label
jks.encrypted = no
jks.keystore_pass = password
jks.key_pass = password
jks.provider = IBMJCE
```

Java JCEKS

```
jceks.keystore = dir/Keystore
jceks.certificate = certificate_label
jceks.encrypted = no
jceks.keystore_pass = password
jceks.key_pass = password
jceks.provider = IBMJCE
```

V 9.1.0 Java JCERACFKS

```
jceracfks.keystore = safkeyring://user/keyring
jceracfks.certificate = certificate_label
```

Tabla 98. Resumen de los parámetros necesarios para cada tipo de archivo de configuración

Parámetros	Obligatorio	Tipo de archivo de configuración			
		V 9.1.0 **Java (JKS, JCEKS y JCERACFKS)**	IBM i **PEM**	**PKCS#11**	**CMS**
keystore	✓	✓			✓
IBM i private	✓		IBM i ✓		
IBM i public	✓		IBM i ✓		
IBM i password	✓		IBM i ✓		
library	✓			✓	
certificate	✓	✓		✓	✓
token	✓			✓	
token_pin	✓			✓	
secondary_keystore	✓			✓	
encrypted		✓			
keystore_pass	✓	✓			

Tabla 98. Resumen de los parámetros necesarios para cada tipo de archivo de configuración (continuación)					
		Tipo de archivo de configuración			
Parámetros	**Obligatorio**	V 9.1.0 **Java (JKS, JCEKS y JCERACFKS)**	IBM i **PEM**	**PKCS#11**	**CMS**
`key_pass`		✓			
`provider`		✓			

Tenga en cuenta que puede añadir comentarios utilizando el símbolo *#* .

Los parámetros del archivo de configuración se definen del modo siguiente:

keystore
Sólo configuración de CMS y Java. Vía de acceso al archivo de claves para la configuración de CMS, JKS y JCEKS.

z/OS V 9.1.0 MQ Adv. VUE URI al archivo de claves RACF para la configuración de JCERACFKS.

Importante:

- La vía de acceso del archivo de almacén de claves no debe incluir la extensión de archivo.
- z/OS V 9.1.0 MQ Adv. VUE El URI al archivo de claves RACF debe tener el formato:

  ```
  safkeyring://user/keyring
  ```

  donde:

  - *user* es el ID de usuario propietario del conjunto de claves
  - *keyring* es el nombre del conjunto de claves.

IBM i **private**
Sólo configuración de PEM. Nombre de un archivo que contiene la clave privada y el certificado en formato PEM.

IBM i **public**
Sólo configuración de PEM. Nombre de un archivo que contiene certificados públicos de confianza en formato PEM.

IBM i **password**
Sólo configuración de PEM. Contraseña que se utiliza para descifrar una clave privada cifrada.

library
Sólo PKCS#11. Nombre de vía de acceso de la biblioteca PKCS#11.

certificate
Sólo configuración de CMS, PKCS#11 y Java. Etiqueta del certificado.

token
Sólo PKCS#11. Etiqueta de señal.

token_pin
Sólo PKCS#11. PIN para desbloquear la señal.

secondary_keystore
Sólo PKCS#11. Nombre de vía de acceso del almacén de claves CMS, que se proporciona sin la extensión `.kdb`, que contiene certificados de ancla (certificados raíz) necesarios para los certificados almacenados en la señal PKCS #11. El almacén de claves secundario también puede contener

certificados que se intermedios en la cadena de confianza, así como certificados de destinatario que se definen en la política de seguridad de privacidad. Este almacén de claves CMS debe ir acompañado de un archivo de ocultación que debe estar ubicado en el mismo directorio que el almacén de claves secundario.

encrypted
Sólo configuración de Java. Estado de la contraseña.

keystore_pass
Sólo configuración de Java. Contraseña del archivo de almacén de claves.

Nota:

- Para el almacén de claves de CMS, AMS depende de archivos de ocultación (`.sth`), mientras que JKS y JCEKS pueden necesitar una contraseña para el certificado y la clave privada del usuario.
- **Importante:** el almacenamiento de contraseñas como texto sin formato supone un riesgo de seguridad.

Nota: Se ignora para `jceracfks`, ya que la contraseña no controla el acceso.

key_pass
Sólo configuración de Java. La contraseña de la clave privada del usuario.

Importante: el almacenamiento de contraseñas como texto sin formato supone un riesgo de seguridad.

Nota: Se ignora para `jceracfks`, ya que la contraseña no controla el acceso.

provider
Sólo configuración de Java. Proveedor de seguridad de Java que aplica los algoritmos criptográficos necesarios para el certificado de almacén de claves.

Importante: la información almacenada en el almacén de claves es esencial para el flujo seguro de los datos enviados utilizando IBM MQ. Los administradores de seguridad deben prestar especial atención al asignar permisos de archivo para estos archivos.

Ejemplo del archivo `keystore.conf`:

```
# Native AMS application configuration
cms.keystore = c:\Documents and Settings\Alice\AliceKeystore
cms.certificate = AliceCert

# Java AMS application configurationjceks.keystore = c:/Documents and Settings/Alice/
AliceKeystore
jceks.certificate = AliceCert
jceks.encrypted = no
jceks.keystore_pass = passw0rd
jceks.key_pass = passw0rd
jceks.provider = IBMJCE
```

Tareas relacionadas

“Protección de contraseñas en Java” en la página 635
El almacenamiento de contraseñas de almacén de claves y de clave privada utilizando texto sin cifrar supone un riesgo de seguridad, así que Advanced Message Security proporciona una herramienta para codificar esas contraseñas utilizando una clave del usuario, que se encuentra en el archivo de almacén de claves.

Soporte para JRE que no son deIBM con AMS

IBM MQ classes for Java y IBM MQ classes for JMS dan soporte a la operación de Advanced Message Security cuando se ejecutan con JRE no IBM.

Advanced Message Security (AMS) implementa Cryptographic Message Syntax (CMS). La sintaxis CMS se utiliza firmar digitalmente, resumir, autenticar o cifrar contenido arbitrario de mensajes.

Desde IBM MQ 9.0, el soporte de Advanced Message Security en IBM MQ classes for Java y IBM MQ classes for JMS utiliza los paquetes Bouncy Castle de código abierto para dar soporte a CMS. Esto significa que estas clases pueden dar soporte a la operación de Advanced Message Security cuando se ejecutan con JRE que no son IBM.

Antes de IBM MQ 9.0, Advanced Message Security no estaba soportado en JRE que no son IBM en clientes Java. El soporte de Advanced Message Security en IBM MQ classes for Java y IBM MQ classes for JMS dependía del soporte de CMS proporcionado específicamente por la implementación de IBM de Java Cryptography Extensions (JCE). Debido a esta restricción, la funcionalidad solo estaba disponible cuando se utilizaba un Java runtime environment (JRE) que incluía el proveedor JCE de Java.

Solaris De manera importante, el soporte en plataformas como, por ejemplo, Solaris requerían un JRE híbrido, es decir, el JRE estándar para la plataforma con elementos adicionales proporcionados por IBM. En concreto, el proveedor de JCE de IBM era necesario, en lugar del proveedor JCE proporcionado por el JRE estándar para la plataforma.

Ubicación y numeración de versiones para archivos JAR de Bouncy Castle

Los archivos JAR de Bouncy Castle que son necesarios para el soporte para los JRE que no son IBM se incluyen como parte del paquete de instalación de IBM MQ classes for Java y IBM MQ classes for JMS.

Los archivos JAR de Bouncy Castle utilizados son los archivos siguientes:

El archivo JAR proporcionado, que es básico para las operaciones de Bouncy Castle.
Este archivo JAR se llama `bcprov-jdk15on.jar`.

El archivo JAR "PKIX", que contiene el soporte para operaciones CMS utilizadas por Advanced Message Security.
Este archivo JAR se llama `bcpkix-jdk15on.jar`.

V 9.1.0.9 **El archivo JAR "util", que contiene las clases utilizadas por los otros archivos JAR de Bouncy Castle.**
Este archivo JAR se llama `bcutil-jdk15on.jar`.

Dependencias

Las clases de IBM MQ 9.1 y clases posteriores se han probado con IBM JRE y Oracle JRE. También es probable que ejecuten correctamente en cualquier JRE compatible con J2SE. Sin embargo, debe tener en cuenta las dependencias siguientes:

- No hay ningún cambio en la configuración de Advanced Message Security.
- Las clases Bouncy Castle se utilizan solo para operaciones de CMS. Todas las demás operaciones relacionadas con la seguridad, por ejemplo, el acceso al almacén de claves, el cifrado real de datos y el cálculo de sumas de comprobación de firma, utilizan la funcionalidad proporcionada por el JRE.

  Importante: Por este motivo, el JRE utilizado debe incluir una implementación del proveedor de JCE.
- Para utilizar algunos algoritmos de cifrado de *alta seguridad*, es posible que tenga que instalar los archivos de política *no restringidos* para la implementación JCE del JRE.

  Consulte la documentación de JRE para obtener más detalles.
- Si ha habilitado la seguridad de Java:
  - Añada `java.security.SecurityPermissioninsertProvider.BC` a la aplicación para que las clases Bouncy Castle se puedan utilizar como proveedor de seguridad.
  - Otorgue `java.security.AllPermission` a los archivos JAR de Bouncy Castle, que son:

    V 9.1.0.9 *mq_install_dir*`/java/lib/bcutil-jdk15on.jar`
    mq_install_dir`/java/lib/bcpkix-jdk15on.jar`

mq_install_dir/java/lib/bcprov-jdk15on.jar

Conceptos relacionados

Qué se instala para las clases de IBM MQ para JMS

Qué se instala para las clases de IBM MQ para Java

Multi Interceptación del agente de canal de mensajes (MCA)

La intercepción MCA permite a un gestor de colas que se ejecuta bajo IBM MQ habilitar de forma selectiva políticas que se van a aplicar para canales de conexión de servidor.

La interceptación de MCA permite a los clientes que permanecen fuera de AMS seguir conectados a un gestor de colas y cifrar y descifrar sus mensajes.

La intercepción MCA se ha diseñado para proporcionar la prestación AMS cuando AMS no se puede habilitar en el cliente. Tenga en cuenta que utilizar la intercepción MCA y un cliente habilitado para AMS lleva a una doble protección que podría ser problemática para recibir aplicaciones. Para obtener más información, consulte "Inhabilitación de Advanced Message Security en el cliente" en la página 624.

Nota: Los interceptores MCA no están soportados para los canales AMQP o MQTT.

Archivo de configuración del almacén de claves

De forma predeterminada, el archivo de configuración de almacén de claves para la interceptación de MCA es `keystore.conf` y se encuentra en el directorio `.mqs` de la vía de acceso del directorio HOME inicial del usuario que ha iniciado el gestor de colas o el escucha. El almacén de claves también se puede utilizar mediante la variable de entorno MQS_KEYSTORE_CONF. Si desea más información sobre cómo configurar el almacén de claves de AMS, consulte "Utilización de almacenes de claves y certificados" en la página 616.

Para habilitar la interceptación de MCA, debe proporcionar el nombre de un canal que desee utilizar en el archivo de configuración de almacén de claves. Para la intercepción MCA, solo se puede utilizar un tipo de almacén de claves cms.

Consulte "Ejemplo de interceptación de MCA de Advanced Message Security" en la página 622 si desea un ejemplo de configuración de la intercepción MCA.

Atención: Debe completar la autenticación de cliente y el cifrado en los canales seleccionados, por ejemplo, utilizando SSL y SSLPEER o CHLAUTH TYPE(SSLPEERMAP), para asegurarse de que solo los clientes autorizados se pueden conectar a y utilizar esta prestación.

IBM i

Si su empresa utiliza IBM iy ha seleccionado una entidad emisora de certificados (CA) comercial para firmar el certificado, el Certificate Manager digital crea una solicitud de certificado en formato PEM (Privacy-Enhanced Mail). Debe enviar la solicitud a su CA elegida.

Para ello, debe utilizar el mandato siguiente para seleccionar el certificado correcto para el canal especificado en `channelname`:

```
pem.certificate.channel.channelname
```

Ejemplo de interceptación de MCA de Advanced Message Security

Una tarea de ejemplo de cómo configurar una interceptación de MCA de AMS.

Antes de empezar

Atención: Debe completar la autenticación de cliente y el cifrado en los canales seleccionados, por ejemplo, utilizando SSL y SSLPEER o CHLAUTH TYPE(SSLPEERMAP), para asegurarse de que solo los clientes autorizados se pueden conectar a y utilizar esta prestación.

Si su empresa utiliza IBM iy ha seleccionado una entidad emisora de certificados (CA) comercial para firmar el certificado, el Certificate Manager digital crea una solicitud de certificado en formato PEM (Privacy-Enhanced Mail). Debe enviar la solicitud a su CA elegida.

Acerca de esta tarea

Esta tarea le guía a través del proceso de configuración del sistema para utilizar la intercepción MCA y, después, de la verificación de la configuración.

Nota: Antes de IBM WebSphere MQ 7.5, AMS era un producto complementario que se tenía que instalar por separado e interceptores configurados para proteger aplicaciones. Desde IBM WebSphere MQ 7.5 hacia adelante, los interceptores se incluyen automáticamente y se habilitan de forma dinámica en los entornos de tiempo de ejecución de cliente y servidor MQ. En este ejemplo de intercepción de MCA, los interceptores se proporcionan en el extremo del canal del servidor, y se utiliza un tiempo de ejecución de cliente más antiguo (en el Paso 12) para colocar mensajes no protegidos en el canal, de forma que se pueda considerar como protegido por los interceptores MCA. Si este ejemplo hubiera utilizado un cliente IBM WebSphere MQ 7.5 o posterior, haría que el mensaje se protegiera dos veces, porque el interceptor de tiempo de ejecución del cliente MQ y el interceptor MCA protegerían el mensaje a medida que entra en MQ.

Atención: Sustituya `userID` en el código por su ID de usuario.

Procedimiento

1. Cree la base de datos de claves y los certificados utilizando los siguientes mandatos para crear un script de shell-

   Además, cambie **INSTLOC** y **KEYSTORELOC** o ejecute los mandatos necesarios. Tenga en cuenta que podría no necesitar crear el certificado para bob.

```
INSTLOC=/opt/mq90
KEYSTORELOC=/home/testusr/ssl/ams1
mkdir -p $KEYSTORELOC
chmod -R 777 $KEYSTORELOC
chown -R mqm:mqm $KEYSTORELOC
export PATH=$PATH:$INSTLOC/gskit8/bin
echo "PATH = $PATH"
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INSTLOC/gskit8/lib64

gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd -stash
gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd -stash
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd
-label alice_cert -dn "cn=alice,O=IBM,c=IN" -default_cert yes
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd
-label bob_cert  -dn "cn=bob,O=IBM,c=IN"  -default_cert yes
```

2. Comparta los certificados entre las dos bases de datos para que cada usuario pueda identificar correctamente al otro.

   Es importante que utilice el método descrito en la tarea 5 de **. Compartir certificados** en la **Guía de inicio rápido** (Windows o UNIX).

3. Cree `keystore.conf` con la siguiente configuración: `Keystore.conf location: /home/userID/ssl/ams1/`

```
cms.keystore = /home/userID/ssl/ams1/alicekey
cms.certificate.channel.SYSTEM.DEF.SVRCONN = alice_cert
```

4. Cree e inicie el gestor de colas AMSQMGR1
5. Defina un escucha con el *puerto* 14567 y el *control* QMGR
6. Inhabilite la autorización de canal o establezca las reglas para la autorización de canal.

   Consulte SET CHLAUTH si desea más información.

7. Detenga el gestor de colas.
8. Establezca el almacén de claves:

```
export MQS_KEYSTORE_CONF=/home/userID/ssl/ams1/keystore.conf
```

9. Inicie el gestor de colas en el mismo shell.
10. Establezca la política de seguridad y 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 y dspmqspl si desea más información.
11. Establezca la configuración de canal:

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/127.0.0.1(14567)'
```

12. Ejecute **amqsputc** desde un cliente MQ que no habilita automáticamente un interceptor MCA; por ejemplo, un cliente IBM WebSphere MQ 7.1 o anterior. Coloque los dos mensajes siguientes:

```
/opt/mqm/samp/bin/amqsputc TESTQ TESTQMGR
```

13. Elimine la política de seguridad y verifique el resultado:

```
setmqspl -m AMSQMGR1 -p TESTQ -remove
dspmqspl -m AMSQMGR1
```

14. Examine la cola desde la instalación de IBM MQ 9.0:

```
/opt/mq90/samp/bin/amqsbcg TESTQ AMSQMGR1
```

    El resultado muestra los mensajes en formato cifrado.
15. Establezca la política de seguridad y verifique el 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. Ejecute **amqsgetc** desde la instalación de IBM MQ 9.0:

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

Tareas relacionadas

“Guía de inicio rápido para AMS con clientes Java” en la página 603
Utilice esta guía para aprender a configurar rápidamente Advanced Message Security para proporcionar seguridad de mensajes para las aplicaciones Java que se conectan utilizando enlaces de cliente. Cuando lo haya completado, habrá creado un almacén de claves para verificar las identidades de usuario y habrá definido políticas de firma/cifrado para su gestor de colas.

Referencia relacionada

“Limitaciones conocidas de AMS” en la página 575
Existe un número de opciones de IBM MQ que no están soportadas o que tienen limitaciones para Advanced Message Security.

Inhabilitación de Advanced Message Security en el cliente

Es necesario inhabilitar IBM MQ Advanced Message Security (AMS) si está utilizando un cliente IBM WebSphere MQ 7.5 o posterior para conectarse a un gestor de colas desde una versión anterior del producto y se notifica un error 2085 （MQRC_UNKNOWN_OBJECT_NAME）.

Acerca de esta tarea

A partir de IBM WebSphere MQ 7.5, IBM MQ Advanced Message Security (AMS) se habilita automáticamente en un cliente de IBM MQ y, por lo tanto, de forma predeterminada, el cliente intenta comprobar las políticas de seguridad para los objetos en el gestor de colas. Sin embargo, los servidores

en versiones anteriores del producto, por ejemplo, IBM WebSphere MQ 7.1, no tienen AMS habilitado y esto provoca que se notifique el error 2085 (MQRC_UNKNOWN_OBJECT_NAME).

Si se notifica este error, cuando intenta conectarse a un gestor de colas desde una versión anterior del producto, puede inhabilitar AMS del modo siguiente:

- Para clientes de Java, en cualquiera de las formas siguientes:
  - Estableciendo una variable de entorno AMQ_DISABLE_CLIENT_AMS.
  - Estableciendo la propiedad del sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS.
  - Utilizando la propiedad DisableClientAMS, bajo la stanza **Security** en el archivo mqclient.ini.
- Para clientes C, de cualquiera de una de las formas siguientes:
  - Estableciendo una variable de entorno MQS_DISABLE_ALL_INTERCEPT.
  - Utilizando la propiedad DisableClientAMS, bajo la stanza **Security** en el archivo mqclient.ini.

Nota: En IBM WebSphere MQ 7.5, también podría utilizar la variable de entorno AMQ_DISABLE_CLIENT_AMS. para clientes C. Desde IBM MQ 8.0, ya no puede utilizar la variable de entorno AMQ_DISABLE_CLIENT_AMS para clientes C. Debe utilizar la variable de entorno MQS_DISABLE_ALL_INTERCEPT en su lugar.

Procedimiento

- Para inhabilitar AMS en el cliente, utilice una de las opciones siguientes:

  Variable de entorno AMQ_DISABLE_CLIENT_AMS
  Es necesario establecer esta variable en los siguientes casos:
  - Si utiliza Java Runtime Environment (JRE) distinto a IBM Java Runtime Environment (JRE)
  - Si está utilizando el cliente IBM WebSphere MQ 7.5 o posterior IBM MQ classes for JMS o IBM MQ classes for Java .

  Cree la variable de entorno AMQ_DISABLE_CLIENT_AMS y establézcala en TRUE en el entorno donde se está ejecutando la aplicación. Por ejemplo:

  ```
  export AMQ_DISABLE_CLIENT_AMS=TRUE
  ```

  La propiedad de sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS
  Para clientes IBM MQ classes for JMS y IBM MQ classes for Java, puede establecer la propiedad del sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS en el valor TRUE para la aplicación Java.

  Por ejemplo, puede establecer la propiedad del sistema Java como una opción -D cuando se invoca el mandato Java:

  ```
  java -Dcom.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS=TRUE -cp <MQ_INSTALLATION_PATH>/java/lib/
  com.ibm.mqjms.jar  my.java.applicationClass
  ```

  De forma alternativa, puede especificar la propiedad del sistema Java dentro de un archivo de configuración de JMS, jms.config, si la aplicación utiliza este archivo.

  Variable de entorno MQS_DISABLE_ALL_INTERCEPT
  Debe establecer esta variable si está utilizando IBM MQ 8.0 o posterior con clientesnativos y debe inhabilitar AMS en el cliente.

  Cree la variable de entorno MQS_DISABLE_ALL_INTERCEPT y establézcala en TRUE en el entorno donde se está ejecutando el cliente. Por ejemplo:

  ```
  export MQS_DISABLE_ALL_INTERCEPT =TRUE
  ```

  Puede utilizar la variable de entorno MQS_DISABLE_ALL_INTERCEPT solo para clientes C. Para clientes Java, debe utilizar la variable de entorno AMQ_DISABLE_CLIENT_AMS en su lugar.

Propiedad `DisableClientAMS` en el archivo `mqclient.ini`

Puede utilizar esta opción para clientes IBM MQ classes for JMS y IBM MQ classes for Java y para clientes C.

Añada el nombre de propiedad `DisableClientAMS` bajo la stanza **`Security`** en el archivo `mqclient.ini`, tal como se muestra en el ejemplo siguiente:

```
Security:
DisableClientAMS=Yes
```

También puede habilitar AMS tal como se indica en el ejemplo siguiente:

```
Security:
DisableClientAMS=No
```

Qué hacer a continuación

Para obtener más información sobre los problemas con la apertura de colas protegidas de AMS , consulte Problemas con la apertura de colas protegidas cuando se utiliza AMS con JMS.

Conceptos relacionados

"Interceptación del agente de canal de mensajes (MCA)" en la página 622
La intercepción MCA permite a un gestor de colas que se ejecuta bajo IBM MQ habilitar de forma selectiva políticas que se van a aplicar para canales de conexión de servidor.

Tareas relacionadas

Configuración de un cliente utilizando un archivo de configuración

Referencia relacionada

El archivo de configuración IBM MQ classes for JMS

Requisitos de certificado para AMS

Los certificados deben tener una clave pública RSA para utilizarlos con Advanced Message Security.

Para obtener más información sobre distintos tipos de clave pública y cómo crearlos, consulte "Certificados digitales y compatibilidad de CipherSpec en IBM MQ" en la página 45.

Extensiones de uso de claves

Las extensiones de uso de claves limitan adicionalmente la forma en la que un certificado se puede utilizar.

En Advanced Message Security, el uso de claves de los certificados X.509 v3 se debe establecer de acuerdo con la especificación RFC 5280.

Para la calidad de la integridad de protección, si se establecen las extensiones de uso de clave de certificado, dicho conjunto debe incluir al menos uno de los dos:

- **`nonRepudiation`**
- **`digitalSignature`**

Para la calidad de la integridad de protección, si se establecen las extensiones de uso de clave de certificado, dicho conjunto debe incluir:

- **`keyEncipherment`**

Para la confidencialidad de la calidad de protección, si se establecen las extensiones de uso de clave de certificado, dicho conjunto debe incluir:

- **`dataEncipherment`**

El uso de claves ampliado limita aún más las extensiones de uso de claves. Para todas las calidades de protección, si se establece el uso de claves ampliado del certificado, el conjunto debe incluir:

- **`emailProtection`**

Conceptos relacionados

"Calidad de protección" en la página 647
Las políticas de protección de datos de Advanced Message Security implican una calidad de protección (QOP).

Métodos de validación de certificados en AMS

Puede utilizar Advanced Message Security para detectar y rechazar los certificados revocados para que los mensajes de las colas no estén protegidos mediante certificados que no cumplen los estándares de seguridad.

AMS le permite verificar la validez de un certificado utilizando Online Certificate Status Protocol (OCSP) o la lista de revocación de certificados (CRL).

AMS se puede configurar para realizar la comprobación mediante OCSP o CRL, o por ambos métodos. Si se habilitan ambos métodos, entonces AMS utiliza primero OCSP para el estado de revocación por motivos de rendimiento. Si el estado de revocación de un certificado es indeterminado después de la comprobación por OCSP, AMS utiliza la comprobación por CRL.

Tenga en cuenta que tanto la comprobación OCSP como la comprobación CRL están habilitadas de forma predeterminada.

Conceptos relacionados

"Protocolo de estado de certificado en línea (OCSP) en AMS" en la página 627
El protocolo de estado de certificado en línea (OCSP) determina si un certificado se ha revocado y, por consiguiente, ayuda a determinar si el certificado puede ser de confianza. De forma predeterminada OCSP está habilitado.

"Listas de revocación de certificados (CRL) en AMS" en la página 629
Las CRL contienen una lista de certificados que han sido marcados por la Autoridad de certificación (CA) como no fiables por diversas razones, por ejemplo, porque la clave privada se ha perdido o está en peligro.

Protocolo de estado de certificado en línea (OCSP) en AMS

El protocolo de estado de certificado en línea (OCSP) determina si un certificado se ha revocado y, por consiguiente, ayuda a determinar si el certificado puede ser de confianza. De forma predeterminada OCSP está habilitado.

OCSP no está soportado en sistemas IBM i.

Habilitación de la comprobación de OCSP para interceptores nativos de Advanced Message Security

La comprobación de OCSP (Online Certificate Status Protocol) en Advanced Message Security está habilitada de forma determinada, según la información de los certificados que se utilizan.

Procedimiento

Añada las opciones siguientes al archivo de configuración del almacén de claves:

Nota: Todo lo incluido en la stanza OCSP es opcional y puede especificarse de forma independiente.

Opción	Descripción
`ocsp.enable=off`	Habilite la comprobación de OCSP si el certificado que se está comprobando tiene una extensión AIA (Authority Info Access) con un método de acceso PKIX_AD_OCSP que contiene un URI de apunta a la ubicación del respondedor de OCSP. Valores posibles: on u `off`.

Opción	Descripción
ocsp.url=*responder_URL*	Dirección de URL del respondedor de OCSP. Si se omite esta opción, la comprobación de OCSP no AIA se inhabilita.
ocsp.http.proxy.host=*OCSP_proxy*	Dirección de URL del servidor proxy de OCSP. Si se omite esta opción, no se utiliza un proxy para las comprobaciones de certificado en línea no de AIA.
ocsp.http.proxy.port=*port_number*	Número de puerto del servidor proxy de OCSP. Si se omite esta opción, se utiliza el puerto predeterminado 8080.
ocsp.nonce.generation=on/off	Generar valor de seguridad al consultar OCSP. El valor predeterminado es off.
ocsp.nonce.check=on/off	Comprobar valor de seguridad después de recibir una respuesta de OCSP. El valor predeterminado es off.
ocsp.nonce.size=8	Tamaño del valor de seguridad, en bytes.
ocsp.http.get=on/off	Especificar HTTP GET como método de solicitud. Si esta opción se establece en off, se utiliza HTTP POST. El valor predeterminado es off.
ocsp.max_response_size=20480	Tamaño máximo de la respuesta proporcionada por el programa de respuesta de OCSP, en bytes.
ocsp.cache_size=100	Habilitar el almacenamiento en memoria caché interna de la respuesta de OCSP y establecer el número límite de entradas de la memoria caché.
ocsp.timeout=30	Tiempo de espera para la respuesta de un servidor, en segundos, pasado el cual Advanced Message Security concluye.
ocsp.unknown=ACCEPT	Defina el comportamiento cuando un servidor OCSP no se puede alcanzar dentro de un periodo de tiempo de espera. Valores posibles: • ACCEPT Permite el certificado • WARN Permite el certificado y registra un aviso • REJECT Impide que el certificado se use y anota un error

Habilitación de la comprobación de OCSP en Java en AMS
Para habilitar la comprobación de OCSP para Java en Advanced Message Security, modifique el archivo java.security o el archivo de configuración del almacén de claves.

Acerca de esta tarea

Existen dos formas de habilitar la comprobación de OCSP en Advanced Message Security:

Utilización de java.security
Compruebe si el certificado contiene una extensión de certificado AIS (Authority Information Access).

Procedimiento

1. Si AIA no está configurado o desea alterar el certificado, edite el archivo $JAVA_HOME/lib/security/java.security con las propiedades siguientes:

   ```
   ocsp.responderURL=http://url.to.responder:port
   ocsp.responderCertSubjectName=CN=Example CA,O=IBM,C=US
   ```

   y habilite la comprobación de OCSP editando el archivo $JAVA_HOME/lib/security/java.security con la línea siguiente:

   ```
   ocsp.enable=true
   ```

2. Si AIA está configurado, habilite la comprobación de OCSP editando el archivo $JAVA_HOME/lib/security/java.security con la línea siguiente:

   ```
   ocsp.enable=true
   ```

Qué hacer a continuación

Si está utilizando Java Security Manager, para completar la configuración, añada el permiso siguiente de Java a lib/security/java.policy

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
```

Utilización de keystore.conf

Procedimiento

Añada el atributo siguiente al archivo de configuración:

```
ocsp.enable=true
```

Importante: Cuando este atributo está definido en el archivo de configuración, prevalece sobre los valores contenidos en java.security.

Qué hacer a continuación

Para completar la configuración, añada los permisos siguientes de Java a lib/security/java.policy:

```
permission java.security.SecurityPermission "getProperty.ocsp.enable";
permission java.security.SecurityPermission "setProperty.ocsp.enable";
```

Listas de revocación de certificados (CRL) en AMS

Las CRL contienen una lista de certificados que han sido marcados por la Autoridad de certificación (CA) como no fiables por diversas razones, por ejemplo, porque la clave privada se ha perdido o está en peligro.

Para validar certificados, Advanced Message Security crea una cadena de certificado que consta del certificado del firmante y el certificado de la entidad emisora de certificados hasta llegar a un ancla de confianza. Un ancla de confianza es un archivo de almacén de confianza que contiene un certificado de confianza o un certificado raíz de confianza que se utiliza para confirmar la confianza de un certificado. AMS verifica la vía de acceso del certificado utilizando un algoritmo de validación PKIX. Cuando se crea y verifica la cadena, AMS completa la validación de certificados, que incluye validar la fecha de emisión y caducidad de cada certificado de la cadena por comparación con la fecha actual y, comprobar si la extensión de uso de la clave está presente en el certificado de entidad final. Si la extensión se añade al

certificado, AMS verifica si **digitalSignature** o **nonRepudiation** también están definidos. Si no lo están, se notifica y registra un error de seguridad MQRC_SECURITY_ERROR. A continuación, AMS descarga listas de revocación de certificados a partir de archivos o de LDAP, dependiendo de los valores que se hayan especificado en el archivo de configuración. AMS sólo es compatible con las listas de revocación de certificados que estén codificadas en el formato DER. Si no se encuentra ninguna configuración de CRL en el archivo de configuración del almacén de claves, AMS no realiza ninguna comprobación de validez por CRL. Para cada certificado de entidad emisora de certificados (certificados de CA), AMS consulta a LDAP para conocer las listas de revocación de certificados (CRL) utilizando nombres distinguidos de una CA para encontrar su CRL. La consulta a LDAP incluye los atributos siguientes:

```
certificateRevocationList,
certificateRevocationList;binary,
authorityRevocationList,
authorityRevocationList;binary
deltaRevocationList
deltaRevocationList;binary,
```

Nota: `deltaRevocationList` sólo se puede utilizar cuando se especifica como puntos de distribución.

Habilitación de la validación de certificados y del soporte de lista de revocación de certificados (CRL) en interceptores nativos
Debe modificar el archivo de configuración del almacén de claves de manera que Advanced Message Security pueda descargar las CRL del servidor LDAP (Lightweight Directory Access Protocol).

Acerca de esta tarea

IBM i La habilitación del soporte de lista de revocación de certificados y de la validación de certificados en los interceptores nativos no está soportado para Advanced Message Security en IBM i.

Procedimiento

Añada las opciones siguientes al archivo de configuración:

Nota: Todo lo incluido en la stanza CRL es opcional y puede especificarse de forma independiente.

Opción	Descripción
`crl.ldap.host=`*`host_name`*	Nombre de host del servidor LDAP.
`crl.ldap.port=`*`port_number`*	Número de puerto del servidor LDAP. Puede especificar un máximo de 11 servidores. Se utilizan varios hosts LDAP para asegurar un relevo transparente en caso de que falle la conexión LDAP. Todos los servidores LDAP son réplicas y contienen los mismos datos. Cuando el interceptor de AMS Java se conecta satisfactoriamente a un servidor LDAP, no intenta descargar CRL de los servidores proporcionados restantes.
`crl.cdp=off`	Utilice esta opción para comprobar o utilizar extensiones CRLDistributionPoints en certificados.
`crl.ldap.version=3`	Número de versión del protocolo LDAP. Valores posibles: 2 ó 3.
`crl.ldap.user=cn=`*`username`*	Inicio de sesión en el servidor LDAP. Si no se especifica este valor, los atributos de CRL en LDAP deben poder ser leídos por todos los usuarios.
`crl.ldap.pass=`*`password`*	Contraseña del servidor LDAP.

Opción	Descripción
`crl.ldap.cache_lifetime=0`	Tiempo de vida de la memoria caché de LDAP, expresado en segundos. Valores posibles: 0-86400.
`crl.ldap.cache_size=50`	Tamaño de la memoria caché de LDAP. Esta opción se puede especificar sólo si el valor de `crl.ldap.cache_lifetime` es mayor que `0`.
`crl.http.proxy.host=some.host.com`	Puerto del servidor proxy HTTP para recuperar CRL CDP.
`crl.http.proxy.port=8080`	Número de puerto del servidor proxy HTTP.
`crl.http.max_response_size=204800`	El tamaño máximo de CRL, en bytes, que se puede recuperar de un servidor HTTP aceptado por GSKit.
`crl.http.timeout=30`	Tiempo de espera para la respuesta de un servidor, en segundos, pasado el cual AMS concluye.
`crl.http.cache_size=0`	Tamaño de la memoria caché de HTTP, en bytes.
`crl.unknown=ACCEPT`	Defina el comportamiento cuando un servidor CRL no se puede alcanzar dentro de un periodo de tiempo de espera. Valores posibles: • `ACCEPT` Permite el certificado • `WARN` Permite el certificado y registra un aviso • `REJECT` Impide que el certificado se use y anota un error

Habilitación del soporte de las listas de revocación de certificados en Java in AMS
Para habilitar las listas de revocación de certificados en Advanced Message Security, debe modificar el archivo de configuración del almacén de claves para permitir que AMS descargue las CRL desde el servidor LDAP (Lightweight Directory Access Protocol) y configurar el archivo java.security.

Procedimiento

1. Añada las opciones siguientes al archivo de configuración:

Cabecera	Descripción
`crl.ldap.host=`*`host_name`*	Nombre de host de LDAP.

Cabecera	Descripción
`crl.ldap.port=`*`port_number`*	Número de puerto del servidor LDAP. Puede especificar un máximo de 11 servidores. Se utilizan varios hosts LDAP para asegurar un relevo transparente en caso de que falle la conexión LDAP. Todos los servidores LDAP son réplicas y contienen los mismos datos. Cuando el interceptor de AMS Java se conecta satisfactoriamente a un servidor LDAP, no intenta descargar CRL de los servidores proporcionados restantes. Java no utiliza los valores `crl.ldap.user` y crl.ldaworldp.pass. Java no utiliza un usuario y una contraseña cuando se conecta a un servidor LDAP. Como consecuencia, los atributos de CRL en LDAP deben poder ser leídos por todos los usuarios.
`crl.cdp=on/off`	Utilice esta opción para comprobar o utilizar extensiones CRLDistributionPoints en certificados.

2. Modifique el archivo JRE/`lib/security/java.security` con las propiedades siguientes:

Nombre de propiedad	Descripción
`com.ibm.security.enableCRLDP`	Esta propiedad toma los valores siguientes: `true`, `false`. Si se establece en `true`, cuando se realiza comprobación de revocación de certificados, las CRL se localizan utilizando el URL de la extensión de puntos de distribución del certificado. Si se establece en `false` o no se establece, se inhabilita la comprobación de CRL mediante la extensión de puntos de distribución de CRL.
`ibm.security.certpath.ldap.cache.lifetime`	Utilice esta propiedad para establecer un valor en segundos para el tiempo de vida de las entradas de la memoria caché del almacén de certificados de LDAP. El valor 0 inhabilita la memoria caché; -1 significa un tipo de vida ilimitado. Si no se define un valor, el tiempo de vida predeterminado es 30 segundos.

Nombre de propiedad	Descripción
`com.ibm.security.enableAIAEXT`	Esta propiedad toma los valores siguientes: `true`, `false`. Si la propiedad se establece en `true`, se examina cualquier extensión de AIA (Authority Information Access) que se encuentre en la vía de acceso del certificados para determinar si contiene los URI de LDAP. Para cada URI de LDAP encontrado, se crea un objeto LDAPCertStore y se añade a la colección de almacenes de certificados que se utiliza para localizar otros certificados que son necesarios para crear la vía de acceso del certificado. Si la propiedad se establece en `false` o no se define, no se crean más objetos LDAPCertStore.

z/OS *Habilitación de las listas de revocación de certificados (CRL) en z/OS*

Advanced Message Security da soporte a la lista de revocación de certificados (CRL) y comprueba los certificados digitales que se utilizan para proteger los mensajes de datos.

Acerca de esta tarea

Cuando está habilitado, Advanced Message Security validará los certificados de destinatarios cuando se colocan mensajes en una cola con protección de privacidad y validarán los certificados de emisor cuando se recuperen los mensajes desde una cola con protección (de integridad o privacidad). En este caso la validación incluye verificar que no se registren certificados relevantes en un CRL relevante.

Advanced Message Security utiliza los servicios de IBM System SSL para validar los certificados de emisores y destinatarios. Puede encontrar la documentación detallada relacionada con la validación de certificados de System SSL en la publicación z/OSCryptographic Services System Secure Sockets Layer Programming (SC24-5901).

Para habilitar la comprobación CRL, especifique la ubicación de un archivo de configuración CRL mediante CRLFILE DD en el JCL de la tarea inicial para el espacio de direcciones AMS. Se proporciona un archivo de configuración CRL de ejemplo en *thlqual*.SCSQPROC(CSQ40CRL). Los valores permitidos en este archivo son los siguientes:

Tabla 99. Variables de configuración CRL de Advanced Message Security

Variable	Valores válidos	Descripción
crl.ldap.host[.n]	*nombrehost -o- nombrehost:puerto*	La direcciónip/nombrehost de su servidor LDAP que aloja los CRL de sus certificados de emisor. Si no especifica un número para su servidor LDAP, se utiliza el número de puerto especificado mediante crl.ldap.port.
crl.ldap.port	*puerto*	El número de puerto TCP/IP de su servidor LDAP.
crl.ldap.user	*usuario_ldap*	El nombre de usuario LDAP que se ha de utilizar para conectarse al servidor LDAP.
crl.ldap.pass	*contraseña_ldap*	La contraseña LDAP asociada a crl.ldap.user.

Puede especificar varios nombres de host y puertos del servidor LDAP como se indica a continuación:

```
crl.ldap.host.1 = hostname -or hostname:port
crl.ldap.host.2 = hostname -or hostname:port
crl.ldap.host.3 = hostname -or hostname:port
```

Puede especificar hasta 10 nombres de host. Si no especifica un número de puerto para sus servidores LDAP, se utiliza el número de puerto especificado mediante crl.ldap.port. Cada servidor LDAP debe utilizar la misma combinación de crl.ldap.user/password para el acceso.

Cuando se especifica CRLFILE DD, se carga la configuración durante la inicialización del espacio de direcciones de Advanced Message Security y se habilita la comprobación de CRL. Si no se especifica CRLFILE DD, o si el archivo de configuración CRL no está disponible o no es válida, se inhabilita la comprobación CRL.

AMS realiza una comprobación CRL utilizando los servicios de validación de certificados de IBM System SSL del modo siguiente:

Tabla 100. Comprobaciones CRL de Advanced Message Security

Operación	**Calidad de protección**	**Certificado(s) comprobado(s)**
PUT	Privacidad	Destinatario(s)
GET	Integridad/Privacidad	Emisor

Si una operación de mensaje no pasa la comprobación CRL, Advanced Message Security realiza las acciones siguientes:

Tabla 101. Comportamiento de error de comprobación CRL de Advanced Message Security

Operación	**Error de comprobación CRL**
PUT	El mensaje no se coloca en la cola de destino. Se devuelve un código de finalización MQCC_FAILED y un código de razón MQRC_SECURITY_ERROR a la aplicación.
GET	Se elimina el mensaje de la cola de destino y se traslada a la cola de errores de protección del sistema. Se devuelve un código de finalización MQCC_FAILED y un código de razón MQRC_SECURITY_ERROR a la aplicación.

AMS para z/OS utiliza los servicios IBM System SSL para validar los certificados, los cuales incluyen la comprobación CRL y la comprobación de confianza. IBM System SSL proporciona la variable de entorno GSK_CRL_SECURITY_LEVEL para moderar el funcionamiento de la comprobación CRL. Por ejemplo:

```
GSK_CRL_SECURITY_LEVEL=MEDIUM
```

Esta variable se documenta en el manual de programación de z/OS Cryptographic Services System Secure Sockets Layer. Las asignaciones válidas incluyen:

- LOW - La validación del certificado no fallará si no se puede contactar con el servidor LDAP.
- MEDIUM - La validación de certificados requiere que se pueda contactar con el servidor LDAP pero no requiere que se haya definido una CRL.
- HIGH - Para la validación de certificados es necesario poder contactar con el servidor LDAP y que haya una CRL definida.

El valor predeterminado de IBM System SSL es MEDIUM. Puede establecer esta variable en el archivo de configuración especificado mediante ENVARS DD en el JCL de tarea iniciada para el espacio de

direcciones AMS. Se proporciona un archivo de configuración de variable de entorno de ejemplo en *thlqual*.SCSQPROC(CSQ40ENV).

Nota: Es responsabilidad del administrador garantizar que los servicios LDAP relevantes estén disponibles y mantener las entradas de CRL para las autoridades certificadoras relevantes.

Protección de contraseñas en Java

El almacenamiento de contraseñas de almacén de claves y de clave privada utilizando texto sin cifrar supone un riesgo de seguridad, así que Advanced Message Security proporciona una herramienta para codificar esas contraseñas utilizando una clave del usuario, que se encuentra en el archivo de almacén de claves.

Antes de empezar

El propietario del archivo `keystore.conf` debe asegurarse de que sólo el propietario del archivo está autorizado para leerlo. La protección de contraseñas que se describe en este capítulo es sólo una medida de protección adicional.

Procedimiento

1. Edite los archivos `keystore.conf` para incluir la vía de acceso del almacén de claves y la etiqueta de usuario.

   ```
   jceks.keystore = c:/Documents and Setting/Alice/AliceKeystore
   jceks.certificate = AliceCert
   jceks.provider = IBMJCE
   ```

2. Para ejecutar la herramienta, emita:

   ```
   java -cp com.ibm.mq.jmqi.jar com.ibm.mq.ese.config.KeyStoreConfigProtector keystore_password
   private_key_password
   ```

   Los datos de salida producidos contienen contraseñas cifradas y se pueden copiar en el archivo `keystore.conf`.

   Para copiar automáticamente los datos de salida en el archivo `keystore.conf`, ejecute:

   ```
   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 obtener una lista de las ubicaciones predeterminadas de `keystore.conf` en distintas plataformas, consulte “Utilización de almacenes de claves y certificados” en la página 616.

Ejemplo

El siguiente es un ejemplo de los datos de salida producidos:

```
#Fri Jul 30 15:20:29 CEST 2010
jceks.key_pass=MMXh997n5ZOr8uRlJmc5qity9MN2CggGBMKCDxdbn1AyPklvdgTsOLG6X3C1YT7oDzwaqZFlOR4t\r\nm
Zsc7JGAx8nqqxLnAucdGn0NWo6xnjZB1n501YGol2k/
PhaQHhFXKMAU9dKg0f8djOtCAOlX4ETe\r\nfYl9LBUt2wk87uM7dSs\=
jceks.keystore_pass=OIdeayBnSCfLG4cFuxEVrk6SYyAsdSPpDqgPf16s9s1M04cqZjNbhgjoA2EXonudHZHH+4s2drvQ
\r\nCUvQgu9GuaBMJK2F2OjtHJJ1Y4BVeLW2c2okgawo/
W2J1AdUYKkJ0raYTkDouLaTYTQeulyG0xIl\r\niD2si1xUCxhYvvyhbbY\=
jceks.encrypted=yes
```

z/OS Utilización de certificados en z/OS

Acerca de esta tarea

Advanced Message Security implementa tres niveles de protección: integridad, confidencialidad y privacidad.

Con una política de integridad, los mensajes se firman utilizando la clave privada del originador (la aplicación que ejecuta MQPUT). La integridad detecta la modificación del mensaje, pero el mensaje propiamente dicho no se cifra.

Con una política de confidencialidad, el mensaje se cifra cuando se coloca en la cola. El mensaje se cifra utilizando una clave simétrica y un algoritmo especificado en la política de Advanced Message Security relevante. La clave simétrica se cifra con la clave pública de cada destinatario (la aplicación que ejecuta MQGET). Las claves públicas se asocian con los certificados almacenados en los conjuntos de claves.

Con una política de privacidad, los mensajes se cifran y también se codifican.

Cuando un mensaje está protegido con privacidad, una aplicación destinataria que ejecuta MQGET lo saca de la cola y el mensaje se debe descifrar. Dado que se ha cifrado utilizando la clave pública del destinatario, se debe descifrar utilizando la clave privada del destinatario que se encuentra en un conjunto de claves.

z/OS *Utilización del conjunto de claves SAF*

Advanced Message Security (AMS) utiliza servicios de conjunto de claves SAF de z/OS para definir y gestionar los certificados necesarios para la firma y el cifrado. Los productos de seguridad que son funcionalmente equivalentes a RACF se pueden utilizar en lugar de RACF si proporcionan el mismo nivel de soporte.

El uso eficaz de los conjuntos de claves puede disminuir la necesidad administrativa de gestionar los certificados.

Una vez generado (o importado) un certificado, éste se debe conectar a un conjunto de claves para que esté accesible. Se puede conectar el mismo certificado a más de un conjunto de claves.

Advanced Message Security utiliza dos grupos de conjuntos de claves. Un grupo consta de los conjuntos de claves propiedad de los ID de usuario individuales que originan o reciben mensajes. Cada conjunto de claves contiene la clave privada asociada al certificado del ID de usuario propietario. La clave privada de cada certificado se utiliza para firmar mensajes para las colas con privacidad protegida o integridad protegida. También se utiliza para descifrar mensajes de colas con la confidencialidad protegida o la privacidad protegida al recibir mensajes.

El otro conjunto es un único conjunto de claves propiedad del usuario del espacio de direcciones de AMS. Contiene la cadena de certificados CA de firmantes necesaria para validar los certificados del originador del mensaje y los destinatarios.

Cuando se utiliza la protección de la privacidad o confidencialidad, el conjunto de claves propiedad del usuario del espacio de direcciones de AMS también contiene los certificados de los destinatarios del mensaje. Las claves públicas de estos certificados se utilizan para cifrar la clave simétrica utilizada para cifrar los datos de mensajes cuando se coloca el mensaje en la cola protegida. Cuando se recuperan estos mensajes, la clave privada de los destinatarios relevantes se utiliza para descifrar la clave simétrica que, a continuación, se utiliza para descifrar los datos del mensaje.

Advanced Message Security utiliza un nombre de conjunto de claves de **drq.ams.keyring** cuando busca certificados y claves privadas. Este es el caso tanto para el usuario, como para los conjuntos de llaves del espacio de direcciones de AMS.

Para ver una ilustración y una descripción detallada de los certificados y el conjunto de claves y su rol en la protección de datos, consulte Resumen de las operaciones relacionadas con los certificados.

La clave privada que se utiliza para la firma y el cifrado puede tener cualquier etiqueta pero debe estar conectada como el certificado predeterminado.

Los certificados digitales y los conjuntos de claves se gestionan en RACF principalmente mediante el uso del mandato RACDCERT.

Si desea más información sobre certificados, etiquetas y el mandato RACDCERT, consulte las publicaciones *z/OS: Security Server RACF Command Language Reference* y *z/OS: Security Server RACF Security Administrator's Guide*.

z/OS *Autorización del acceso al mandato RACDCERT*

La autorización para utilizar el mandato RACDCERT es una tarea posterior a la instalación que debe completar el programador del sistema z/OS. Esta tarea requiere conceder los permisos necesarios al administrador de seguridad de Advanced Message Security.

Resumiendo, estos mandatos son necesarios para permitir el acceso al mandato RACF RACDCERT:

```
RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID( admin ) ACCESS(CONTROL)
SETROPTS RACLIST(FACILITY) REFRESH
```

En este ejemplo, *admin* especifica el ID de usuario del administrador de seguridad o cualquier usuario que desee que utilice el mandato RACDCERT.

z/OS *Creación de certificados y conjuntos de claves*

Esta sección documenta los pasos necesarios para crear los certificados y los conjuntos de claves necesarios para los usuarios z/OS de Advanced Message Security (AMS), utilizando una autoridad de certificado (CA) de RACF.

Resolución de problemas con certificados cuando se utiliza Advanced Message Security en z/OS

Si tiene problemas con certificados y faltan entradas en los almacenes de claves, puede habilitar un rastreo GSKIT.

En el archivo referenciado por el ENVARS DD en el procedimiento de tarea iniciada de AMS. añada:

```
GSK_TRACE_FILE=/u/... /gsktrace
GSK_TRACE=0xff
```

Consulte Variables de entorno para obtener más información.

Para cada acceso al almacén de claves, los datos se graban en el archivo de rastreo especificado en GSK_TRACE_FILE.

Para dar formato al archivo de rastreo, use el mandato:

```
gsktrace inputtrace file > output_file
```

Caso en particular

Para describir los pasos necesarios, se utiliza un caso de ejemplo de una aplicación emisora y una aplicación receptora.

En los ejemplos siguientes, user1 es el originador de un mensaje y user2 es el destinatario. El ID de usuario del espacio de direcciones de Advanced Message Security es WMQAMSD.

Todos los mandatos de los ejemplos que se muestran aquí se emiten desde la opción 6 de ISPF mediante el ID de usuario administrativo admin.

z/OS *Definición de un certificado de la autoridad certificadora local*

Si está utilizando RACF como su autoridad certificadora (CA), debe crear un certificado de autoridad certificado, si todavía no lo ha hecho. El mandato que se muestra aquí crea un certificado de una autoridad certificadora (o firmante). En este ejemplo se crea un certificado denominado AMSCA que se utilizará cuando se creen los certificados siguientes que reflejen la identidad de los usuarios y las aplicaciones de Advanced Message Security.

Este mandato se puede modificar, especialmente SUBJECTSDN, para que refleje la estructura de nombres y los convenios que utiliza su instalación:

```
RACDCERT CERTAUTH GENCERT SUBJECTSDN(CN('AMSCA') O('ibm') C('us'))
KEYUSAGE(CERTSIGN) WITHLABEL('AMSCA')
```

Nota: Los certificados firmados con este certificado de autoridad certificadora local muestran un emisor CN=AMSCA,O=ibm,C=us cuando se listan con el mandato RACDCERT LIST.

z/OS *Creación de un certificado digital con una clave privada*

Se debe generar un certificado digital con un clave privada para cada usuario de Advanced Message Security. En el ejemplo siguiente, se utilizan los mandatos RACDCERT para generar certificados para los usuarios user1 y user2, los cuales están firmados por el certificado CA local identificado mediante la etiqueta AMSCA.

```
RACDCERT ID(user1) GENCERT SUBJECTSDN(CN('user1') O('ibm') C('us'))
WITHLABEL('user1') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(user2) GENCERT SUBJECTSDN(CN('user2') O('ibm') C('us'))
WITHLABEL('user2') SIGNWITH(CERTAUTH LABEL('AMSCA'))
KEYUSAGE(HANDSHAKE DATAENCRYPT DOCSIGN)

RACDCERT ID(user1) ALTER (LABEL('user1')) TRUST
RACDCERT ID(user2) ALTER (LABEL('user2')) TRUST
```

El mandato RACDCERT ALTER es necesario para añadir el atributo TRUST al certificado. Cuando se crea un certificado por primera vez utilizando este procedimiento, su rango de fechas válidas es diferente del rango del certificado firmante. Como resultado, RACF lo marca como NOTRUST, lo que significa que el certificado no se va a utilizar. Utilice el mandato RACDCERT ALTER para establecer el atributo TRUST.

Se deben especificar atributos de KEYUSAGE, HANDSHAKE, DATAENCRYPT y DOCSIGN para los certificados que utiliza Advanced Message Security.

Tabla 102. Valores de RACDCERT KEYUSAGE e indicadores

Valor de KEYUSAGE	**Conjunto de indicadores**
HANDSHAKE	digitalSignature y keyEncipherment
DATAENCRYPT	dataEncipherment
DOCSIGN	nonRepudiation
CERTSIGN	keyCertSign y cRLSign

z/OS *Creación de conjuntos de claves RACF*

Los mandatos que se muestran aquí crean un conjunto de claves para los ID de usuario definidos porRACF, user1, user2, y el usuario de la tarea del espacio de direcciones de Advanced Message Security WMQAMSD. El nombre del conjunto de claves lo fija Advanced Message Security y debe codificarse tal como se muestra, sin comillas. El nombre es sensible a las mayúsculas y minúsculas.

```
RACDCERT ID(user1) ADDRING(drq.ams.keyring)
```

```
RACDCERT ID(user2) ADDRING(drq.ams.keyring)
RACDCERT ID(WMQAMSD) ADDRING(drq.ams.keyring)
```

z/OS *Conexión de certificados a los conjuntos de claves*

Conecte los certificados de usuario y de la CA a los conjuntos de claves:

```
RACDCERT ID(WMQAMSD) CONNECT(CERTAUTH LABEL('AMSCA')
RING(drq.ams.keyring))
RACDCERT ID(user1) CONNECT(ID(user1) LABEL('user1')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(user2) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) DEFAULT USAGE(PERSONAL))
RACDCERT ID(WMQAMSD) CONNECT(ID(user2) LABEL('user2')
RING(drq.ams.keyring) USAGE(SITE))
```

El certificado que contiene la clave privada utilizada para el cifrado se debe conectar al conjunto de claves como el certificado predeterminado.

El atributo RACDCERT USAGE(SITE) impide que la clave privada esté accesible en el conjunto de claves, mientras que el atributo RACDCERT USAGE(PERSONAL) permite utilizar la clave privada, si existe. El certificado del User2 debe estar conectado al conjunto de claves del espacio de direcciones de Advanced Message Security porque su clave pública es necesaria para cifrar mensajes, a medida que se colocan en la cola. USAGE(SITE) limita la exposición de la clave privada de user2.

El certificado CERTAUTH con la etiqueta AMSCA se debe conectar al conjunto de claves del espacio de direcciones de Advanced Message Security ya que se ha utilizado para firmar el certificado de user1, el cual es el originador del mensaje. Se utiliza para validar el certificado firmado de user1.

z/OS *Verificación del conjunto de claves*

El conjunto de claves debe aparecer como se muestra aquí, después de que se hayan especificado todos los mandatos:

```
RACDCERT ID(user1) LISTRING(drq.ams.keyring)
Digital ring information for user USER1:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
user1                            ID(USER1)    PERSONAL YES

RACDCERT ID(user2) LISTRING(drq.ams.keyring)
Digital ring information for user USER2:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
user2                            ID(USER2)    PERSONAL YES

RACDCERT ID(WMQAMSD) LISTRING(drq.ams.keyring)
Digital ring information for user WMQAMSD:
Ring:>drq.ams.keyring<:

Certificate Label Name           Cert Owner   USAGE    DEFAULT
------------------------------ ------------ -------- -------
AMSCA                            CERTAUTH     CERTAUTH NO
user2                            ID(USER2)    SITE     NO
```

El listado de los certificados individuales también muestra la asociación del conjunto de claves.

```
RACDCERT ID(user2) LIST(label('user2'))
Digital certificate information for user USER2:

***
Label: user2
Certificate ID: 2QfH8Pny9/LzpKKFmfFA
Status: TRUST
Start Date: 2010/05/03 22:59:53
```

```
End Date:  2011/05/04 22:59:52
Serial Number:>15<:
Issuer's Name:>OU=AMSCA.O=ibm.C=us<:
Subject's Name:>CN=user2.O=ibm.C=us<:
Key Usage: HANDSHAKE, DATAENCRYPT, DOCSIGN
Private Key Type: Non-ICSF
Private Key Size: 1024
Ring Associations:
Ring Owner: USER2
Ring:>drq.ams.keyring<:
Ring Owner: WMQAMSD
Ring:>drq.ams.keyring<:
```

Para mejorar el rendimiento, el contenido del drq.ams.keyring asociado al espacio de direcciones de AMS se almacena en la memoria caché para la vida útil del espacio de direcciones. Los cambios en dicho conjunto de claves no entran en vigor de forma automática. El administrador puede renovar la memoria caché realizando lo siguiente:

- Detener y reiniciar el gestor de colas.
- Utilización del mandato MODIFY de z/OS:

```
F qmgrAMSM,REFRESH KEYRING
```

Tareas relacionadas

Operando Advanced Message Security

z/OS *Resumen de las operaciones relacionadas con certificados*

La Figura 35 en la página 640 ilustra las relaciones entre el envío y la recepción de las aplicaciones y los certificados relevantes. El caso de uso ilustrado implica la gestión de colas remotas entre dos gestores de colas de z/OS que utilizan una política de privacidad de protección de datos. En Figura 35 en la página 640, "AMS" indica " Advanced Message Security".

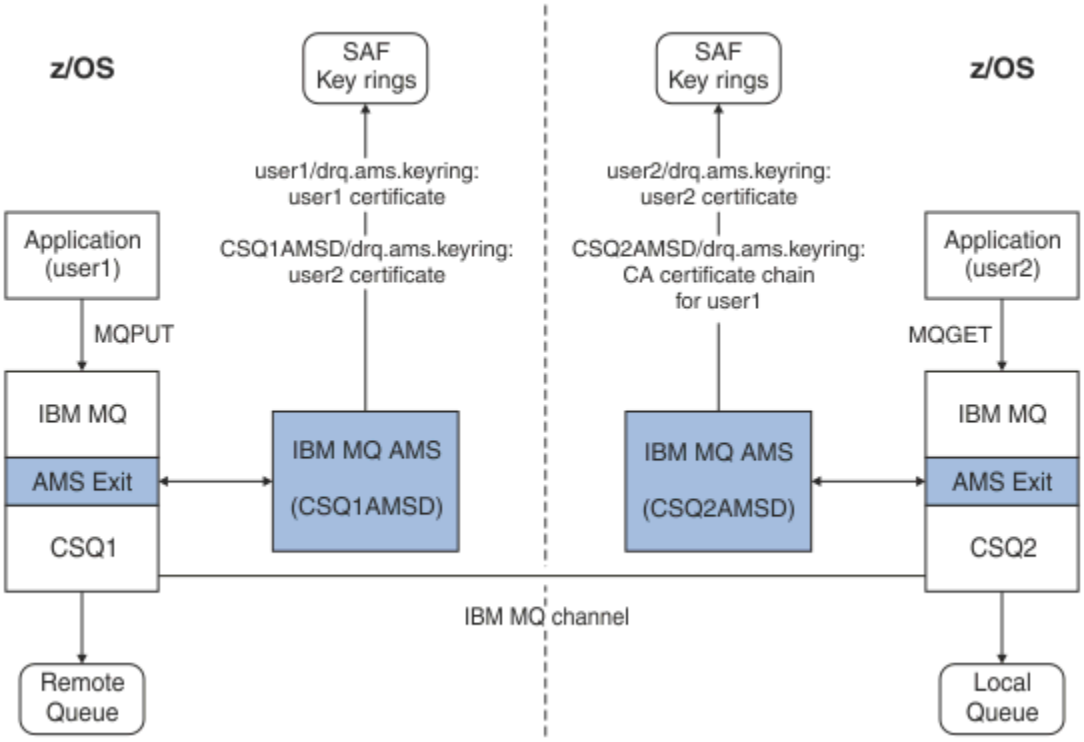


Figura 35. Relaciones entre las aplicaciones y los certificados

En este diagrama, una aplicación que se ejecuta como 'user1' coloca un mensaje en una cola remota gestionada por el gestor de colas CSQ1, con la intención de que lo recupere una aplicación que se ejecuta

como 'user2' desde una cola local gestionada por el gestor de colas CSQ2. El diagrama presupone una política de privacidad Advanced Message Security, lo que significa que el mensaje está firmado y cifrado.

Advanced Message Security intercepta el mensaje cuando éste se coloca y utiliza el certificado de user2 (almacenado en el conjunto de claves del usuario en el espacio de direcciones AMS) para cifrar una clave simétrica que se utiliza para cifrar los datos del mensaje.

Tenga en cuenta que el certificado de user2 está conectado al conjunto de claves del espacio de direcciones AMS con la opción USAGE(SITE). Esto significa que el usuario del espacio de direcciones AMS puede acceder al certificado y a la clave pública pero no a la clave privada.

En el extremo de recepción, Advanced Message Security intercepta la obtención que ha emitido user2 y utiliza el certificado de user2 para descifrar la clave simétrica de modo que pueda descifrar los datos del mensaje. A continuación, valida la firma de user1 utilizando la cadena de certificados de CA del certificado de user1 almacenado en el conjunto de claves del usuario del espacio de direcciones AMS.

En este caso de uso, pero con una política de integridad de protección de datos, no serían necesarios los certificados para user2.

Para utilizar Advanced Message Security para colocar en cola los mensajes de las colas protegidas por IBM MQ que tienen una política de privacidad o integridad de protección de datos, Advanced Message Security debe tener acceso a estos elementos de datos:

- El certificado X.509 V2 o V3 y la clave privada del usuario que está colocando el mensaje en la cola.
- La cadena de certificados utilizada para firmar los certificados digitales de todos los firmantes de mensajes.
- Si la política de protección de datos es de privacidad, el certificado X.509 V2 o V3 de los destinatarios previstos. Los destinatarios previstos figuran listados en la política de Advanced Message Security asociada a la cola.

Para los procesos y aplicaciones que se ejecutan en z/OS, Advanced Message Security debe tener certificados en dos ubicaciones:

- En un conjunto de claves gestionado por ASM con la identidad RACF de la aplicación emisora (la aplicación que coloca en cola el mensaje protegido) o la aplicación receptora (si se utiliza la privacidad).

  El certificado que Advanced Message Security localiza es el certificado predeterminado y debe incluir la clave privada. Advanced Message Security asume la identidad de usuario de z/OS de la aplicación emisora. Es decir, actúa como un sustituto, de modo que puede acceder a la clave privada del usuario.
- En un conjunto de claves gestionado por SAF con el usuario del espacio de direcciones AMS.

  Cuando envía mensajes protegidos por privacidad, este conjunto de claves contiene los certificados de claves públicas de los destinatarios de los mensajes. Cuando recibe mensajes, contiene la cadena de certificados de la autoridad certificadora necesaria para validar la firma del emisor del mensaje.

En los ejemplos anteriores se ha utilizado RACF como la CA local. Sin embargo, puede utilizar otro proveedor de PKI (autoridad certificadora) en su instalación. Si tiene previsto utilizar otro producto de PKI, recuerde que la clave privada y el certificado se deben importar a un conjunto de claves asociado al ID de usuario de z/OS RACF que origina los mensajes IBM MQ protegidos por Advanced Message Security.

Puede utilizar el mandato RACDCERT de RACF como mecanismo para generar solicitudes de certificados, que se pueden exportar y enviar al proveedor de PKI de su elección para que se emitan.

El siguiente es un resumen de los pasos relacionados con el certificado:

1. Solicite la creación de un certificado de CA, uno en el que RACF sea la CA local. Omita este paso si está utilizando otro proveedor de PKI.
2. Genere certificados de usuario firmados por la CA.
3. Cree los conjuntos de claves para los usuarios y el ID del espacio de direcciones AMS de Advanced Message Security.
4. Conecte el certificado de usuario al conjunto de claves de usuario con el atributo predeterminado.

5. Conecte los certificados de destinatarios al conjunto de claves de usuario del espacio de direcciones AMS de Advanced Message Security utilizando el atributo usage(site). Este paso es necesario únicamente para los certificados de usuario que finalmente serán los destinatarios de los mensajes protegidos mediante privacidad.
6. Conecte las cadenas de certificados de la CA para los emisores de mensajes al conjunto de claves de usuario del espacio de direcciones AMS de Advanced Message Security. Este paso es necesario únicamente para las tareas AMS que verificarán las firmas del emisor.

z/OS *Configuración de una PKI no residente de z/OS*

Advanced Message Security para z/OS, utiliza certificados digitales X.509 V3 en el proceso de protección de los mensajes que se colocan en o reciben de colas de IBM MQ. Advanced Message Security no crea ni gestiona el ciclo de vida de estos certificados por su cuenta; esta función la proporciona una infraestructura de claves públicas (PKI). Los ejemplos de esta publicación que ilustran el uso de los certificados utilizan z/OS Security Server RACF para cumplimentar solicitudes de certificados.

Tanto si utiliza una PKI residente de z/OS o no de z/OS, AMS para z/OS solo utiliza los conjuntos de claves gestionados por RACF o su equivalente. Estos conjuntos de claves están basados en SAF (Security Authorization Facility) son el repositorio utilizado por AMS para z/OS para recuperar certificados para los emisores y receptores de los mensajes que se colocan o reciben desde las colas de IBM MQ.

Para los mensajes emitidos desde z/OS, que están protegidos por la política de integridad o cifrado, el certificado y la clave privada del ID de usuario de origen se debe almacenar en un conjunto de claves gestionado por SAF que esté asociado al ID de usuario de z/OS que ha originado el mensaje.

RACF incluye la posibilidad de importar certificados y claves privadas en conjuntos de claves gestionados por RACF-managed key rings. Consulte las publicaciones de z/OS Security Server RACF para obtener información detallada y ejemplos sobre cómo cargar certificados en los conjuntos de claves gestionados por RACF managed key rings.

Si su instalación utiliza uno de los productos PKI soportados, consulte las publicaciones que se incluyen con el producto para obtener información acerca de su despliegue.

Administración de políticas de seguridad de Advanced Message Security

Advanced Message Security utiliza políticas de seguridad para especificar los algoritmos criptográficos de cifrado y firma para cifrar y autenticar los mensajes que fluyen a través de las colas.

Visión general de las políticas de seguridad para AMS

Las políticas de seguridad de Advanced Message Security son objetos conceptuales que describen la forma en que un mensaje se cifra y se firma criptográficamente.

Para obtener más detalles sobre los atributos de política de seguridad, consulte los temas subordinados siguientes:

Conceptos relacionados

“Calidad de protección” en la página 647
Las políticas de protección de datos de Advanced Message Security implican una calidad de protección (QOP).

“Atributos de política de seguridad en AMS” en la página 646
Puede utilizar Advanced Message Security para seleccionar un algoritmo o método determinados para proteger los datos.

Nombres de política en AMS

El nombre de política es un nombre exclusivo que identifica una determinada política de Advanced Message Security y la cola a la que se aplica.

El nombre de política debe ser el mismo que el nombre de la cola a la que se aplica. Existe una correlación unívoca entre una política de Advanced Message Security (AMS) y una cola.

Al crear una política con el mismo nombre que el de una cola, se activa la política para dicha cola. Las colas sin nombres de política coincidentes no están protegidas por AMS.

El ámbito de la política es relevante para el gestor de colas local y sus colas. Los gestores de colas remotos deben tener sus propias políticas definidas localmente para las colas que gestionan.

Algoritmo de firma en AMS

El algoritmo de firma indica el algoritmo que se debe utilizar al firmar mensajes de datos.

Los valores válidos incluyen:

- MD5
- SHA-1
- Familia SHA-2:
  - SHA256
  - SHA384 (longitud de clave mínima aceptable: 768 bits)
  - SHA512 (longitud de clave mínima aceptable: 768 bits)

Una política que no especifica un algoritmo de firma, o especifica un algoritmo de NONE, implica que los mensajes colocados en la cola asociada a la política no están firmados.

Nota: La calidad de protección utilizada para colocar y recuperar mensajes debe coincidir. Si existe una discrepancia en la calidad de protección de la política entre la cola y el mensaje de la cola, el mensaje no se acepta y se envía a la cola de manejo de errores. Esta regla es válida tanto para colas locales como remotas.

Algoritmo de cifrado en AMS

El algoritmo de cifrado indica el algoritmo que debe utilizarse al cifrar los mensajes de datos colocados en la cola asociada a la política.

Los valores válidos incluyen:

- RC2
- DES
- 3DES
- AES128
- AES256

Una política que no especifica un algoritmo de cifrado o especifica un algoritmo de NONE implica que los mensajes colocados en la cola asociada a la política no están cifrados.

Tenga en cuenta que una política que especifica un algoritmo de cifrado que no sea NONE también debe especificar como mínimo un nombre distinguido de destinatario y un algoritmo de firma porque los mensajes cifrados de Advanced Message Security también están firmados.

Importante: La calidad de protección utilizada para colocar y recuperar mensajes debe coincidir. Si existe una discrepancia en la calidad de protección de la política entre la cola y el mensaje de la cola, el mensaje no se acepta y se envía a la cola de manejo de errores. Esta regla es válida tanto para colas locales como remotas.

Tolerancia en AMS

El atributo de tolerancia indica si Advanced Message Security puede aceptar mensajes sin ninguna política de seguridad especificada.

Cuando se recupera un mensaje de una cola con una política para cifrar mensajes, si el mensaje no está cifrado, se devuelve a la aplicación de llamada. Los valores válidos incluyen:

0
No (**valor predeterminado**).

1
Sí.

Una política que no especifica un valor de tolerancia o especifica 0 implica que los mensajes colocados en la cola asociada a la política deben coincidir con las reglas de política.

La tolerancia es opcional y existe para facilitar el despliegue de la configuración cuando se han aplicado políticas a las colas, pero esas colas ya contienen mensajes que no tienen una política de seguridad especificada.

Nombres distinguidos de emisor en AMS

Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola. Un remitente utiliza su certificado para firmar un mensaje, antes de colocar el mensaje en una cola.

Advanced Message Security (AMS) no comprueba si un usuario válido ha colocado un mensaje en una cola de datos protegidos hasta que se recupera el mensaje. En este momento, si la política estipula uno o más remitentes válidos, y el usuario que ha colocado el mensaje en la cola no está en la lista de remitentes válidos, AMS devuelve un error a la aplicación receptora y coloca el mensaje en la cola de errores AMS.

Una política puede tener 0 o más nombres distinguidos de emisor válidos. Si no se especifica ningún DN de remitente para la política, cualquier remitente puede colocar mensajes protegidos por datos en la cola siempre que se confíe en el certificado del remitente. Un certificado del remitente es de confianza añadiendo el certificado público a un almacén de claves disponible para la aplicación receptora.

Los nombres distinguidos de emisor tienen el siguiente formato:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Todos los DN deben estar en mayúsculas. Todos los identificadores de nombre de componente del DN deben especificarse en el orden que se muestra en la tabla siguiente:

Nombre de componente	Valor
CN	Nombre común del objeto de este nombre distinguido, tal como un nombre completo o el uso previsto de un dispositivo.
OU	Unidad dentro de la organización a la que está afiliado el objeto del nombre distinguido, tal como un departamento empresarial o un nombre de producto.
O	Organización a la que está afiliado el objeto del nombre distinguido, tal como una empresa.
L	Localidad (ciudad o municipio) donde está situado el objeto del nombre distinguido.
ST	Nombre del estado o provincia donde está situado el objeto del nombre distinguido.
C	País donde está situado el objeto del nombre distinguido.

- Si se especifica uno o más DN de emisor para la política, sólo dichos usuarios pueden colocar mensajes en la cola asociada con la política.
- Los DN de emisor, cuando se especifican, deben coincidir exactamente con el DN contenido en el certificado digital asociado con el usuario que coloca el mensaje.
- AMS permite utilizar nombres distinguidos con caracteres pertenecientes solamente al conjunto de caracteres Latin-1 Para crear DN con caracteres del conjunto, primero debe crear un certificado con un DN que se crea en la codificación UTF-8 utilizando UNIX con la codificación UTF-8 activada o con la GUI

de **strmqikm** . A continuación, debe crear una política desde una plataforma UNIX con la codificación UTF-8 activada o utilizar el conector de AMS con IBM MQ.

- El método utilizado por AMS para convertir el nombre del remitente del formato x.509 al formato de DN siempre utiliza ST= para el valor de estado o provincia.
- Los siguientes caracteres especiales necesitan caracteres de escape:

```
, (comma)
+ (plus)
" (double quote)
\ (backslash)
< (less than)
> (greater than)
; (semicolon)
```

- Si el nombre distinguido contiene blancos intercalados, debe especificarlo entre comillas dobles.

Conceptos relacionados

"Nombres distinguidos de destinatario en AMS" en la página 645
Los nombres distinguidos de destinatario identifican a usuarios que están autorizados a recuperar mensajes de una cola.

Nombres distinguidos de destinatario en AMS

Los nombres distinguidos de destinatario identifican a usuarios que están autorizados a recuperar mensajes de una cola.

Una política puede tener 0 o más nombres distinguidos de destinatario válidos. Los nombres distinguidos de destinatario tienen el formato siguiente:

```
CN=Common Name,O=Organization,C=Country
```

Importante:

- Todos los DN deben estar en mayúsculas. Todos los identificadores de nombre de componente del DN deben especificarse en el orden que se muestra en la tabla siguiente:

Nombre de componente	Valor
CN	Nombre común del objeto de este nombre distinguido, tal como un nombre completo o el uso previsto de un dispositivo.
OU	Unidad dentro de la organización a la que está afiliado el objeto del nombre distinguido, tal como un departamento empresarial o un nombre de producto.
O	Organización a la que está afiliado el objeto del nombre distinguido, tal como una empresa.
L	Localidad (ciudad o municipio) donde está situado el objeto del nombre distinguido.
ST	Nombre del estado o provincia donde está situado el objeto del nombre distinguido.
C	País donde está situado el objeto del nombre distinguido.

- Si no se especifica ningún DN de destinatario para la política, cualquier usuario podrá obtener mensajes de la cola asociada con la política.
- Si se especifica uno o más DN de destinatario para la política, sólo dichos usuarios pueden obtener mensajes de la cola asociada con la política.

- Los DN de destinatario, cuando se especifican, deben coincidir exactamente con el DN contenido en el certificado digital asociado con el usuario que obtiene el mensaje.
- Advanced Message Security permite utilizar nombres distinguidos con caracteres pertenecientes solamente al conjunto de caracteres Latin-1 Para crear DN con caracteres del conjunto, primero debe crear un certificado con un DN que se crea en la codificación UTF-8 utilizando UNIX con la codificación UTF-8 activada o con la GUI de **strmqikm** . A continuación, debe crear una política desde una plataforma UNIX con la codificación UTF-8 activada o utilizar el conector de Advanced Message Security con IBM MQ.

Conceptos relacionados

"Nombres distinguidos de emisor en AMS" en la página 644
Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola. Un remitente utiliza su certificado para firmar un mensaje, antes de colocar el mensaje en una cola.

Atributos de política de seguridad en AMS

Puede utilizar Advanced Message Security para seleccionar un algoritmo o método determinados para proteger los datos.

Una política de seguridad es un objeto conceptual que describe la forma en que un mensaje se cifra y firma criptográficamente.

Tabla 103. Atributos de política de seguridad en AMS

Atributos	**Descripción**
Nombre de política	Nombre exclusivo de la política para un gestor de colas.
Algoritmo de firma	Algoritmo criptográfico que se utiliza para firmar mensajes antes de enviarlos.
Algoritmo de cifrado	Algoritmo criptográfico que se utiliza para cifrar mensajes antes de enviarlos.
Lista de destinatarios	Lista de nombres distinguidos (DN) de certificado de posibles receptores de un mensaje.
Lista de comprobación de nombres distinguidos de firma	Lista de nombres distinguidos de firma que se deben validar durante la recuperación de mensajes.

En Advanced Message Security, los mensajes se cifran con una clave simétrica, y la clave simétrica se cifra con las claves públicas de los destinatarios. Las claves públicas se cifran con el algoritmo RSA, con claves que tienen una longitud efectiva máxima de 2048 bits. El cifrado de clave asimétrica real depende de la longitud de la clave de certificado.

Los algoritmos de clave simétrica que se pueden utilizar son los siguientes:

- RC2
- DES
- 3DES
- AES128
- AES256

Advanced Message Security también puede utilizar las funciones hash criptográficas siguientes:

- MD5
- SHA-1
- Familia SHA-2:
  - SHA256

- SHA384 (longitud de clave mínima aceptable: 768 bits)
- SHA512 (longitud de clave mínima aceptable: 768 bits)

Nota: La calidad de protección utilizada para colocar y recuperar mensajes debe coincidir. Si existe una discrepancia en la calidad de protección de la política entre la cola y el mensaje de la cola, el mensaje no se acepta y se envía a la cola de manejo de errores. Esta regla es válida tanto para colas locales como remotas.

Calidad de protección

Las políticas de protección de datos de Advanced Message Security implican una calidad de protección (QOP).

Los tres niveles de calidad de protección en Advanced Message Security se complementan con un cuarto nivel en IBM MQ 9.0 y posterior, y todo depende de los algoritmos criptográficos que se utilizan para firmar y cifrar el mensaje.

- Privacidad - los mensajes colocados en la cola deben estar firmados y cifrados.
- Integridad - los mensajes colocados en la cola deben estar firmados por el emisor.
- Confidencialidad - los mensajes colocados en la cola deben estar cifrados. Para obtener más información, consulte “Calidades de protección disponibles con AMS” en la página 572
- Ninguna - no se aplica ninguna protección de datos.

Una política que establece que los mensajes deben estar firmados cuando se colocan en una cola tiene una calidad de protección INTEGRITY. La calidad de protección INTEGRITY significa que una política estipula un algoritmo de firma, pero no estipula un algoritmo de cifrado. Los mensajes protegidos por integridad se denominan también mensajes firmados ("SIGNED").

Una política que establece que los mensajes deben estar firmados y cifrados cuando se colocan en una cola tiene una calidad de protección PRIVACY. La calidad de protección PRIVACY significa que una política estipula un algoritmo de firma y un algoritmo de cifrado. Los mensajes protegidos por privacidad se denominan también mensajes sellados ("SEALED").

Una política que establece que los mensajes deben estar cifrados cuando se colocan en una cola tiene una calidad de protección CONFIDENTIALITY. La calidad de protección CONFIDENTIALITY significa que una política estipula un algoritmo de cifrado.

Una política que no estipula un algoritmo de firma ni un algoritmo de cifrado tiene una calidad de protección NONE. Advanced Message Security no proporciona ninguna protección de datos para las colas que tienen una política cuya calidad de protección es NONE.

Gestión de políticas de seguridad

Una política de seguridad es un objeto conceptual que describe la forma en que un mensaje se cifra y firma criptográficamente.

La ubicación desde la que se ejecutan todas las tareas administrativas relacionadas con las políticas de seguridad depende de la plataforma que se utiliza.

- ULW En UNIX, y Windows, utilice los mandatos DELETE POLICY, DISPLAY POLICY y SET POLICY (o PCF equivalentes) para gestionar las políticas de seguridad.
  - UNIX En UNIX, las tareas administrativas se pueden ejecutar desde *MQ_INSTALLATION_PATH*/bin.
  - Windows En las plataformas Windows, las tareas administrativas se pueden ejecutar desde cualquier ubicación, ya que las variables de entorno PATH se actualizan durante la instalación.
- IBM i En IBM i, los mandatos DSPMQMSPL, SETMQMSPL y WRKMQMSPL se instalan en la biblioteca del sistema QSYS para el idioma principal del sistema cuando se instala IBM MQ.

  Las versiones traducidas adicionales se instalan en bibliotecas QSYS29xx de acuerdo con la carga de características de idioma. Por ejemplo, una máquina que tiene inglés de Estados Unidos como idioma

principal y coreano como idioma secundario tiene instalados los mandatos en inglés de Estados Unidos en QSYS y la carga de idioma secundario coreano en QSYS2962 ya que 2962 es la carga de idioma para coreano.

- z/OS En z/OS, los mandatos administrativos se ejecutan utilizando el programa de utilidad de política de seguridad de mensajes (CSQ0UTIL). Cuando se crean, modifican o suprimen políticas en z/OS, Advanced Message Security no reconoce los cambios hasta que se detiene y reinicia o se utiliza el mandato MODIFY de z/OS para renovar la configuración de políticas de Advanced Message Security. Por ejemplo:

```
F <qmgr ssid>AMSM,REFRESH POLICY
```

Tareas relacionadas

“Creación de políticas de seguridad en AMS” en la página 648
Las políticas de seguridad definen la forma en que se protege un mensaje cuando se coloca en una cola o cuando se recibe.

“Modificación de políticas de seguridad en AMS” en la página 649
Puede utilizar Advanced Message Security para modificar los detalles de las políticas de seguridad que ya ha definido.

“Visualización y volcado de las políticas de seguridad en AMS” en la página 650
Utilice el mandato **dspmqspl** para visualizar una lista de todas las políticas de seguridad o detalles de una política con nombre de acuerdo con los parámetros que proporcione en la línea de mandatos.

“Eliminación de políticas de seguridad en AMS” en la página 651
Para eliminar las políticas de seguridad en Advanced Message Security, debe utilizar el mandato `setmqspl`.

Operando Advanced Message Security

Referencia relacionada

El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL)

Creación de políticas de seguridad en AMS

Las políticas de seguridad definen la forma en que se protege un mensaje cuando se coloca en una cola o cuando se recibe.

Antes de empezar

Existen algunas condiciones básicas que se deben cumplir al crear las políticas de seguridad:

- El gestor de colas debe estar en ejecución.
- El nombre de una política de seguridad debe seguir las Reglas para denominar objetos de IBM MQ.
- Debe tener la autorización necesaria para conectarse al gestor de colas y crear una política de seguridad:
  - z/OS En z/OS, otorgue las autorizaciones documentadas en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).
  - Multi En otras plataformas distintas a z/OS, debe conceder las autorizaciones +connect, +inq y +chg necesarias mediante el mandato setmqaut.

  Para obtener más información acerca de cómo configurar la seguridad, consulte el apartado “Configuración de seguridad” en la página 128.
- z/OS En z/OS, asegúrese de que los objetos del sistema necesarios se hayan definido de acuerdo con las definiciones contenidas en CSQ4INSM.

Ejemplo

A continuación se muestra un ejemplo de creación de una política en el gestor de colas QMGR. La política especifica que los mensajes se firmen utilizando el algoritmo SHA256 y se cifren utilizando el algoritmo AES256 para los certificados con DN: CN=joe,O=IBM,C=US y DN: CN=jane,O=IBM,C=US. Esta política está asociada a MY.QUEUE:

```
setmqspl -m QMGR -p MY.QUEUE -s SHA256 -e AES256 -r CN=joe,O=IBM,C=US -r CN=jane,O=IBM,C=US
```

A continuación se muestra un ejemplo de creación de una política en el gestor de colas QMGR. La política especifica que los mensajes se cifren utilizando el algoritmo 3DES para los certificados con los nombres distinguidos: CN=john, O=IBM,C=US y CN=jeff,O=IBM,C=US y firmados con el algoritmo SHA256 para el certificado con el nombre distinguido: CN=phil,O=IBM,C=US

```
setmqspl -m QMGR -p MY.OTHER.QUEUE -s SHA256 -e 3DES -r CN=john,O=IBM,C=US -r
CN=jeff,O=IBM,C=US -a CN=phil,O=IBM,C=US
```

Nota:

- La calidad de protección utilizada para colocar y recuperar mensajes debe coincidir. Si la calidad de protección de la política que se ha definido para el mensaje es más débil que la definida para una cola, el mensaje se envía a la cola de manejo de errores. Esta política es válida tanto para colas locales como remotas.

Referencia relacionada

Lista completa de los atributos del mandato setmqspl

Modificación de políticas de seguridad en AMS

Puede utilizar Advanced Message Security para modificar los detalles de las políticas de seguridad que ya ha definido.

Antes de empezar

- El gestor de colas con el que desee trabajar debe estar en ejecución.
- Debe tener la autorización necesaria para conectarse al gestor de colas y crear una política de seguridad.
  - z/OS En z/OS, otorgue las autorizaciones documentadas en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).
  - Multi En otras plataformas distintas a z/OS, debe conceder las autorizaciones +connect, +inq y +chg necesarias mediante el mandato setmqaut.

  Para obtener más información acerca de cómo configurar la seguridad, consulte el apartado “Configuración de seguridad” en la página 128.

Acerca de esta tarea

Para cambiar las políticas de seguridad, aplique el mandato `setmqspl` a una política ya existente proporcionando nuevos atributos.

Ejemplo

A continuación se muestra un ejemplo de creación de una política denominada MYQUEUE en un gestor de colas denominado QMGR, que especifica que los mensajes se van a cifrar utilizando el algoritmo 3DES para autores (**-a**) que tienen certificados con el nombre distinguido (DN) CN=alice, O=IBM, C=US y firmado con el algoritmo SHA256 para destinatarios (**-r**) que tienen certificados con el DN CN=jeff, O=IBM, C = US.

```
setmqspl -m QMGR -p MYQUEUE -e 3DES -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Para modificar esta política, emita el mandato `setmqspl` con todos los atributos del ejemplo cambiando sólo los valores que desea modificar. En este ejemplo, una política creada previamente se asocia a una nueva cola y su algoritmo de cifrado se cambia a AES256:

```
setmqspl -m QMGR -p MYQUEUE -e AES256 -s SHA256 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Referencia relacionada

setmqspl (establecer política de seguridad)

Visualización y volcado de las políticas de seguridad en AMS

Utilice el mandato **dspmqspl** para visualizar una lista de todas las políticas de seguridad o detalles de una política con nombre de acuerdo con los parámetros que proporcione en la línea de mandatos.

Antes de empezar

- Para visualizar los detalles de las políticas de seguridad, el gestor de colas debe existir y estar en ejecución.
- Debe tener la autorización necesaria para conectarse al gestor de colas y crear una política de seguridad.
  - z/OS En z/OS, otorgue las autorizaciones documentadas en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).
  - Multi En otras plataformas distintas a z/OS, debe conceder las autorizaciones +connect, +inq y +chg necesarias mediante el mandato setmqaut.

  Para obtener más información acerca de cómo configurar la seguridad, consulte el apartado “Configuración de seguridad” en la página 128.

Acerca de esta tarea

A continuación se muestra una lista de los distintivos del mandato **dspmqspl**:

*Tabla 104. Distintivos del mandato **dspmqspl**.*

Distintivo del mandato	Explicación
-m	Nombre del gestor de colas (obligatorio).
-p	Nombre de política.
-export	La adición de este distintivo genera datos de salida que se pueden aplicar fácilmente a un gestor de colas diferente.

Ejemplo

En el ejemplo siguiente se muestra cómo crear dos políticas de seguridad para `venus.queue.manager`:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s sha256 -a "CN=signer1,O=IBM,C=US" -e NONE
setmqspl -m venus.queue.manager -p AMS_POL_06_THREE -s sha256 -a "CN=another signer,O=IBM,C=US"
-e NONE
```

Este ejemplo muestra un mandato que muestra detalles de todas las políticas definidas para `venus.queue.manager` y el resultado que produce:

```
dspmqspl -m venus.queue.manager

Policy Details:
Policy name: AMS_POL_04_ONE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
```

```
Encryption algorithm: NONE
Signer DNs:
  CN=signer1,O=IBM,C=US
Recipient DNs: -
Toleration: 0
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

Este ejemplo muestra un mandato que muestra detalles de una política de seguridad seleccionada definida para `venus.queue.manager` y el resultado que produce:

```
dspmqspl -m venus.queue.manager -p AMS_POL_06_THREE

Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: SHA256
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

En el ejemplo siguiente, en primer lugar debemos crear una política de seguridad y, a continuación, exportar la política utilizando el distintivo **`-export`**:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s SHA256 -a "CN=signer1,O=IBM,C=US" -e NONE

dspmqspl -m venus.queue.manager -export
```

z/OS En z/OS, la información de la política exportada la graba CSQ0UTIL en EXPORT DD.

Multi En plataformas distintas a z/OS, redireccione la salida a un archivo, por ejemplo:

```
dspmqspl -m venus.queue.manager -export > policies.[bat|sh]
```

Para importar una política de seguridad:

- Windows En Windows, ejecute `policies.bat`.
- UNIX En UNIX:
  1. Inicie la sesión como un usuario que pertenece al grupo de administración `mqm` IBM MQ.
  2. Emita `. policies.sh`.
- z/OS En z/OS utilice el programa de utilidad CSQ0UTIL, especificando en SYSIN el conjunto de datos que contiene la información de la política exportada.

Referencia relacionada

Lista completa de los atributos del mandato dspmqspl

Eliminación de políticas de seguridad en AMS

Para eliminar las políticas de seguridad en Advanced Message Security, debe utilizar el mandato `setmqspl`.

Antes de empezar

Existen algunas condiciones básicas que se deben cumplir al gestionar las políticas de seguridad:

- El gestor de colas debe estar en ejecución.
- Debe tener la autorización necesaria para conectarse al gestor de colas y crear una política de seguridad.
  - z/OS En z/OS, otorgue las autorizaciones documentadas en El programa de utilidad de política de seguridad de mensajes (CSQ0UTIL).
  - Multi En otras plataformas distintas a z/OS, debe conceder las autorizaciones +connect, +inq y +chg necesarias mediante el mandato setmqaut.

  Para obtener más información acerca de cómo configurar la seguridad, consulte el apartado “Configuración de seguridad” en la página 128.

Acerca de esta tarea

Utilice el mandato **setmqspl** con la opción **-remove**.

Ejemplo

A continuación se muestra un ejemplo de eliminación de una política:

```
setmqspl -m QMGR -remove -p MY.OTHER.QUEUE
```

Referencia relacionada

Lista completa de los atributos del mandato setmqspl

Protección de colas del sistema en AMS

Las colas del sistema permiten la comunicación entre IBM MQ y sus aplicaciones auxiliares. Cada vez que se crea un gestor de colas, se crea también una cola del sistema para almacenar mensajes y datos internos de IBM MQ. Puede proteger colas del sistema con Advanced Message Security para que solamente los usuarios autorizados puedan acceder a ellas o descifrarlas.

La protección de colas del sistema sigue el mismo patrón que la protección de colas normales. Consulte “Creación de políticas de seguridad en AMS” en la página 648.

Windows Para utilizar la protección de colas del sistema en Windows, copie el archivo `keystore.conf` en el directorio siguiente:

```
c:\Documents and Settings\Default User\.mqs\keystore.conf
```

z/OS En z/OS, para proporcionar protección para `SYSTEM.ADMIN.COMMAND.QUEUE`, el servidor de mandatos debe tener acceso a `keystore` y `keystore.conf`, que contienen claves y una configuración para que el servidor de mandatos pueda acceder a claves y certificados. Todos los cambios realizados en la política de seguridad de `SYSTEM.ADMIN.COMMAND.QUEUE` requieren reiniciar el servidor de mandatos.

Todos los mensajes que se intercambian con la cola de mandatos se firman o se firman y cifran dependiendo de los valores de la política. Si un administrador define firmantes autorizados, el servidor de mandatos no ejecuta los mensajes de mandatos que no pasan la comprobación de nombre distinguido (DN) del firmante y no se direccionan a la cola de manejo de errores de Advanced Message Security. Los mensajes que se envían como respuestas a colas dinámicas temporales de IBM MQ Explorer no están protegidos por AMS.

Las políticas de seguridad no afectan a las colas del sistema siguientes:

- SYSTEM.ADMIN.ACCOUNTING.QUEUE
- SYSTEM.ADMIN.ACTIVITY.QUEUE
- SYSTEM.ADMIN.CHANNEL.EVENT

- SYSTEM.ADMIN.COMMAND.EVENT
- z/OS SYSTEM.ADMIN.COMMAND.QUEUE
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.LOGGER.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.PUBSUB.EVENT
- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.STATISTICS.QUEUE
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE
- SYSTEM.AUTH.DATA.QUEUE
- SYSTEM.BROKER.ADMIN.STREAM
- z/OS SYSTEM.BROKER.CLIENTS.DATA
- SYSTEM.BROKER.CONTROL.QUEUE
- SYSTEM.BROKER.DEFAULT.STREAM
- SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS
- z/OS SYSTEM.BROKER.SUBSCRIPTIONS.DATA
- SYSTEM.CHANNEL.INITQ
- SYSTEM.CHANNEL.SYNCQ
- z/OS SYSTEM.CHLAUTH.DATA.QUEUE
- SYSTEM.CICS.INITIATION.QUEUE
- SYSTEM.CLUSTER.COMMAND.QUEUE
- SYSTEM.CLUSTER.HISTORY.QUEUE
- SYSTEM.CLUSTER.REPOSITORY.QUEUE
- SYSTEM.CLUSTER.TRANSMIT.QUEUE
- z/OS SYSTEM.COMMAND.INPUT
- z/OS SYSTEM.DDELAY.LOCAL.QUEUE
- SYSTEM.DEAD.LETTER.QUEUE
- SYSTEM.DURABLE.SUBSCRIBER.QUEUE
- SYSTEM.HIERARCHY.STATE
- SYSTEM.INTER.QMGR.CONTROL
- SYSTEM.INTER.QMGR.FANREQ
- SYSTEM.INTER.QMGR.PUBS
- SYSTEM.INTERNAL.REPLY.QUEUE
- z/OS SYSTEM.JMS.PS.STATUS.QUEUE
- z/OS SYSTEM.JMS.REPORT.QUEUE
- SYSTEM.PENDING.DATA.QUEUE
- SYSTEM.PROTECTION.ERROR.QUEUE
- SYSTEM.PROTECTION.POLICY.QUEUE
- z/OS SYSTEM.QSG.CHANNEL.SYNCQ
- z/OS SYSTEM.QSG.TRANSMIT.QUEUE

- z/OS SYSTEM.QSG.UR.RESOLUTION.QUEUE
- SYSTEM.RETAINED.PUB.QUEUE
- z/OS SYSTEM.RETAINED.PUB.QUEUE
- SYSTEM.SELECTION.EVALUATION.QUEUE
- SYSTEM.SELECTION.VALIDATION.QUEUE

Cómo otorgar permisos OAM

Los permisos de archivos autorizan a todos los usuarios ejecutar los mandatos `setmqspl` y `dspmqspl`. Sin embargo, Advanced Message Security depende del Gestor de autorizaciones sobre objetos (OAM) y todo intento de ejecutar estos mandatos por un usuario que no pertenezca al grupo mqm, que es el grupo de administración de IBM MQ, o que no tenga permisos para leer los valores de política de seguridad que se otorgan, da como resultado un error.

Procedimiento

Para otorgar los permisos necesarios a un usuario, ejecute:

```
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
```

Nota: Sólo es necesario establecer estas autorizaciones de OAM si tiene previsto conectar clientes, al gestor de colas, utilizando Advanced Message Security 7.0.1.

Atención: Autorización para examinar SYSTEM.PROTECTION.POLICY.QUEUE no es obligatorio en todas las situaciones. IBM MQ optimiza el rendimiento almacenando en memoria caché las políticas para que no tenga que examinar los registros para obtener detalles de política en el SYSTEM.PROTECTION.POLICY.QUEUE en todos los casos.

IBM MQ no almacena en memoria caché todas las políticas disponibles. Si hay un número alto de políticas, IBM MQ almacena en memoria caché un número limitado de políticas. Por lo tanto, si el gestor de colas tiene un número bajo de políticas definidas, no es necesario proporcionar la opción de examinar a SYSTEM.PROTECTION.POLICY.QUEUE.

Sin embargo, debe otorgar autorización para examinar esta cola, en caso de que haya un gran número de políticas definidas, o si está utilizando clientes antiguos. El sistema SYSTEM.PROTECTION.ERROR.QUEUE se utiliza para colocar mensajes de error generados por el código AMS. La autorización de colocación sobre esta cola sólo se comprueba cuando se intenta colocar un mensaje de error en la cola. La autorización de colocación sobre la cola no se comprueba cuando intenta transferir u obtener un mensaje de una cola protegida por AMS.

Concesión de permisos de seguridad

Cuando utilice la seguridad de recursos de mandatos, debe configurar los permisos para permitir que Advanced Message Security funcione. En este tema se utilizan mandatos RACF en los ejemplos. Si su empresa utiliza un gestor de seguridad externo (ESM) diferente, debe utilizar los mandatos equivalentes para dicho ESM.

Existen tres aspectos para conceder permisos de seguridad:

- "El espacio de direcciones AMSM" en la página 655
- "CSQ0UTIL" en la página 655
- "Utilización de colas que tienen definida una política de Advanced Message Security" en la página 655

Notas: Los mandatos de ejemplo utilizan las variables siguientes.

1. *NombreGestColas*: El nombre del gestor de colas.

z/OS En z/OS, este valor puede ser el nombre de un grupo de compartición de colas.

2. *nombre_usuario*: Este valor puede ser un nombre de grupo.
3. Los ejemplos muestran la clase MQQUEUE. También puede ser MXQUEUE, GMQQUEUE o GMXQUEUE. Consulte "Perfiles para la seguridad de colas" en la página 201 para obtener más información.

Además, si ya existe el perfil, no necesita el mandato RDEFINE.

El espacio de direcciones AMSM

Necesita emitir alguna seguridad de IBM MQ para el nombre de usuario bajo el que se ejecuta el espacio de direcciones de Advanced Message Security.

- Para la conexión por lotes con el gestor de colas, emita:

```
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
           PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Para el acceso a SYSTEM.PROTECTION.POLICY.QUEUE, emita:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
           PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

CSQOUTIL

El programa de utilidad que permite a los usuarios ejecutar mandatos **setmqspl** y **dspmqspl** requiere los siguientes permisos, donde el nombre de usuario es el ID de usuario del trabajo:

- Para la conexión por lotes con el gestor de colas, emita:

```
  RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
            PERMIT QMgrName.BATCH CLASS(MQCONN) ID(username) ACCESS(READ)
```

- Para acceder a SYSTEM.PROTECTION.POLICY.QUEUE, lo cual es necesario para el mandato **setmqpol**, emita:

```
  RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
              PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(ALTER)
```

- Para acceder a SYSTEM.PROTECTION.POLICY.QUEUE, lo cual es necesario para el mandato **dspmqpol**, emita:

```
  RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
               PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

Utilización de colas que tienen definida una política de Advanced Message Security

Cuando una aplicación no realiza ningún trabajo con colas que tienen definida una política, dicha aplicación requiere permisos adicionales para permitir que Advanced Message Security proteja los mensajes.

La aplicación requiere:

- Acceso de lectura a SYSTEM.PROTECTION.POLICY.QUEUE. Esto se lleva a cabo, emitiendo:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)
          PERMIT QMgrName.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

- Acceso de transferencia a SYSTEM.PROTECTION.ERROR.QUEUE. Esto se lleva a cabo, emitiendo:

```
RDEFINE MQQUEUE QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE UACC(NONE)
           PERMIT QMgrName.SYSTEM.PROTECTION.ERROR.QUEUE CLASS(MQQUEUE)
ID(username) ACCESS(READ)
```

IBM i Configuración de certificados y el archivo de configuración del almacén de claves en IBM i

La primera tarea al configurar la protección de Advanced Message Security es crear un certificado y asociarlo con el entorno. La asociación se configura a través de un archivo retenido en el sistema de archivos integrado (IFS).

Procedimiento

1. Para crear un certificado autofirmado utilizando el conjunto de herramientas OpenSSL que se entrega con IBM i, emita el siguiente mandato desde QShell:

   ```
   /QOpenSys/usr/bin/openssl req -x509 -newkey rsa:2048 -keyout
   $HOME/private.pem -out $HOME/mycert.pem -nodes -days 365
   ```

   El mandato solicita diversos atributos de nombre distinguido para un nuevo certificado autofirmado, incluidos:

   - Common Name (CN=)
   - Organization (O=)
   - Country (C=)

   Esto crea una clave privada sin cifrar y un certificado coincidente, los dos en formato PEM (Privacy Enhanced Mail).

   Para simplificar, especifique solo los valores para el nombre común, organización y país. Estos atributos y valores son importantes al crear una política.

   Las solicitudes y los atributos adicionales se pueden personalizar especificando un archivo de configuración openssl personalizado en la línea de mandatos con el parámetro **-config**. Consulte la documentación de OpenSSL para obtener más detalles sobre la sintaxis del archivo de configuración.

   Por ejemplo, el mandato siguiente añade extensiones adicionales de certificado X.509 v3:

   ```
   /QOpenSys/usr/bin/openssl req -x509 -newkey rsa:2048
   -keyout $HOME/private.pem -out $HOME/mycert.pem -nodes -days 365 -config myconfig.cnf
   ```

   donde myconfig.cnf es un archivo de corriente ASCII que contiene lo siguiente:

   ```
   [req]
   distinguished_name = req_distinguished_name
   x509_extensions = myextensions

   [req_distinguished_name]
   countryName = Country Name (2 letter code)
   countryName_default = GB
   stateOrProvinceName = State or Province Name (full name)
   stateOrProvinceName_default = Hants
   localityName = Locality Name (eg, city)
   localityName_default = Hursley
   organizationName = Organization Name (eg, company)
   organizationName_default = IBM United Kingdom
   organizationalUnitName = Organizational Unit Name (eg, department)
   organizationalUnitName_default = IBM MQ Development
   commonName = Common Name (eg, Your Name)

   [myextensions]
   keyUsage = digitalSignature,nonRepudiation,dataEncipherment,keyEncipherment
   extendedKeyUsage = emailProtection
   ```

2. AMS requiere que tanto el certificado como la clave privada se mantengan en el mismo archivo. Emita el siguiente mandato para hacerlo:

```
cat $HOME/mycert.pem >> $HOME/private.pem
```

   El archivo `private.pem` en $HOME ahora contiene una clave privada y un certificado coincidentes, mientras que el archivo `mycert.pem` contiene todos los certificados públicos para los que puede cifrar mensajes y validar firmas.

   Es necesario asociar los dos archivos con su entorno creando un archivo de configuración de almacén de claves, `keystore.conf`, en la ubicación predeterminada.

   De forma predeterminada, AMS busca la configuración del almacén de claves en un subdirectorio `.mqs` del directorio de inicio.
3. En QShell, cree el archivo `keystore.conf`:

```
mkdir -p $HOME/.mqs
echo "pem.private = $HOME/private.pem" > $HOME/.mqs/keystore.conf
echo "pem.public = $HOME/mycert.pem" >> $HOME/.mqs/keystore.conf
echo "pem.password = unused" >> $HOME/.mqs/keystore.conf
```

IBM i *Creación de una política en IBM i*

Antes de crear una política, es necesario crear una cola para retener los mensajes protegidos.

Procedimiento

1. En un indicador de línea de mandatos escriba:

```
CRTMQMQ QNAME(PROTECTED) QTYPE(*LCL) MQMNAME (mqmname)
```

   donde mqmname es el nombre del gestor de colas.

   Utilice el mandato DSPMQM para comprobar que el gestor de colas es capaz de utilizar políticas de seguridad. Asegúrese de que **Security Policy Capability** muestra **YES*.

   La política más simple que puede definir es una política de integridad, que se consigue creando una política con un algoritmo de firma digital, pero sin algoritmo de cifrado.

   Los mensajes están firmados pero no cifrados. Si los mensajes se van a cifrar, debe especificar un algoritmo de cifrado y uno o más destinatarios de mensajes que desee.

   Un certificado del almacén de claves público para un destinatario de mensaje que desee se identifica mediante un nombre distinguido.
2. Visualice los nombres distinguidos de los certificados en el almacén de claves público, `mycert.pem` en $HOME, utilizando el siguiente mandato en QShell:

```
/QOpenSys/usr/bin/openssl x509 -in $HOME/mycert.pem -noout -subject -nameopt RFC2253
```

   Es necesario especificar el nombre distinguido como un destinatario deseado y el nombre de política debe coincidir con el nombre de cola que debe protegerse.
3. Por ejemplo, en un indicador de mandatos CL escriba:

```
SETMQMSPL POLICY(PROTECTED) MQMNAME (mqmname)SIGNALG(*SHA256) ENCALG(*AES256) RECIP('CN=..,
O=.., C=..')
```

   donde mqmname es el nombre del gestor de colas.

   Una vez que se haya creado la política, todos los mensajes que se colocan, examinan o eliminan de modo destructivo a través de dicho nombre de cola están sujetos a la política de AMS.

Referencia relacionada

Visualizar gestor de colas de mensajes (DSPMQM)

Establecer política de seguridad de MQM (SETMQMSPL)

IBM i *Probar la política en IBM i*

Utilice las aplicaciones de ejemplo proporcionadas con el producto para probar las políticas de seguridad.

Acerca de esta tarea

Puede utilizar las aplicaciones de ejemplo proporcionadas con IBM MQ, como AMQSPUT4, AMQSGET4, AMQSGBR4 y las herramientas como WRKMQMMSG para transferir, examinar y obtener mensajes utilizando el nombre de cola PROTECTED.

Siempre y cuando todo se haya configurado correctamente, no debería haber ninguna diferencia en el comportamiento de la aplicación con el de una cola no protegida para este usuario.

Sin embargo, un usuario no configurado para Advanced Message Security o un usuario que no tenga la clave privada necesaria para descifrar el mensaje, no podrá ver el mensaje. El usuario recibe un código de terminación de RCFAIL, equivalente a MQCC_FAILED (2) y un código de razón de RC2063 (MQRC_SECURITY_ERROR).

Para ver que la protección AMS está en vigor, transfiera algunos mensajes de prueba a la cola PROTECTED, por ejemplo utilizando AMQSPUT0. Entonces podrá crear una cola de alias para examinar los datos protegidos sin formato mientras se está en reposo.

Procedimiento

Para otorgar los permisos necesarios a un usuario, ejecute:

```
CRTMQMQ QNAME(ALIAS) QTYPE(*ALS) TGTQNAME(PROTECTED) MQMNAME(yourqm)
```

Examinar utilizando el nombre de cola ALIAS, por ejemplo utilizando AMQSBCG4 o WRKMQMMSG, debería revelar mensajes `scrambled` más grandes donde un examen de la cola PROTECTED muestra mensajes de texto simple.

Los mensajes mezclados son visibles pero el texto simple original no puede descifrarse utilizando la cola ALIAS, ya que no hay política para que AMS fuerce la coincidencia de este nombre. Por lo tanto, se devuelven los datos protegidos sin formato.

Referencia relacionada

Establecer política de seguridad de MQM (SETMQMSPL)

Trabajar con mensajes MQ (WRKMQMMSG)

Sucesos de mandato y de configuración

Con Advanced Message Security, puede generar mensajes para sucesos de mandato y de configuración, que se pueden registrar y servir como registro de los cambios de política con fines de auditoría.

Los sucesos de mandatos y configuración que genera IBM MQ son mensajes en formato PCF enviados a las colas dedicadas del gestor de colas donde se produce el suceso.

Los mensajes de sucesos de configuración se envían a la cola SYSTEM.ADMIN.CONFIG.EVENT.

Los mensajes de sucesos de mandatos se envían a la cola SYSTEM.ADMIN.COMMAND.EVENT.

Los sucesos se generan con independencia de las herramientas que utilice para gestionar las políticas de seguridad de Advanced Message Security.

En Advanced Message Security, existen cuatro tipos de sucesos generados por distintas acciones en políticas de seguridad:

- “Creación de políticas de seguridad en AMS” en la página 648, que produce dos mensajes de suceso de IBM MQ:
  - Un suceso de configuración
  - Un suceso de mandato
- “Modificación de políticas de seguridad en AMS” en la página 649, que produce tres mensajes de suceso de IBM MQ:
  - Un suceso de configuración que contiene valores antiguos de política de seguridad
  - Un suceso de configuración que contiene valores nuevos de política de seguridad
  - Un suceso de mandato
- “Visualización y volcado de las políticas de seguridad en AMS” en la página 650, que produce un solo mensajes de suceso de IBM MQ:
  - Un suceso de mandato
- “Eliminación de políticas de seguridad en AMS” en la página 651, que produce dos mensajes de suceso de IBM MQ:
  - Un suceso de configuración
  - Un suceso de mandato

Habilitación e inhabilitación del registro de sucesos

Puede controlar sucesos de mandato y de configuración mediante los atributos del gestor de colas **CONFIGEV** y **CMDEV**. Para habilitar estos sucesos, establezca el atributo de gestor de colas adecuado en ENABLED. Para inhabilitar estos sucesos, establezca el atributo adecuado del gestor de colas en DISABLED.

Procedimiento

Sucesos de configuración

Para habilitar los sucesos de configuración, establezca **CONFIGEV** en ENABLED. Para inhabilitar los sucesos de configuración, establezca **CONFIGEV** en DISABLED. Por ejemplo, puede habilitar los sucesos de configuración mediante el mandato MQSC siguiente:

```
ALTER QMGR CONFIGEV (ENABLED)
```

Sucesos de mandatos

Para habilitar los sucesos de mandatos, establezca **CMDEV** en ENABLED. Para habilitar los sucesos de mandato para todos los mandatos, excepto los mandatos **DISPLAY MQSC** y los mandatos Inquire PCF, establezca **CMDEV** en NODISPLAY. Para inhabilitar los sucesos de mandato, establezca **CMDEV** en DISABLED. Por ejemplo, puede habilitar los sucesos de mandato mediante el mandato MQSC siguiente:

```
ALTER QMGR CMDEV (ENABLED)
```

Tareas relacionadas

Control de sucesos de configuración, mandato y registro en IBM MQ

Formato del mensaje de suceso de mandato

El mensaje de suceso de mandato consta de la estructura MQCFH y los parámetros PCF que le siguen a continuación.

Estos son valores de MQCFH seleccionados:

```
Type = MQCFT_EVENT;
Command = MQCMD_COMMAND_EVENT;
MsgSeqNumber = 1;
```

```
Control = MQCFC_LAST;
ParameterCount = 2;
CompCode = MQCC_WARNING;
Reason = MQRC_COMMAND_PCF;
```

Nota: El valor de ParameterCount es dos porque siempre hay dos parámetros de tipo MQCFGR (grupo). Cada grupo consta de parámetros adecuados. Los datos de suceso constan de dos grupos, CommandContext y CommandData.

CommandContext contiene:

EventUserID

Descripción: El ID de usuario que ha emitido el mandato o la llamada que ha generado el suceso. (Éste es el mismo ID de usuario que se utiliza para comprobar la autorización para poder emitir el mandato o la llamada; para los mandatos recibidos de una cola, éste es también el identificador de usuario (UserIdentifier) del MD del mensaje de mandato).

Identificador: MQCACF_EVENT_USER_ID.

Tipo de datos: MQCFST.

Longitud máxima: MQ_USER_ID_LENGTH.

Se devuelve: Siempre.

EventOrigin

Descripción: El origen de la acción que ha provocado el suceso.

Identificador: MQIACF_EVENT_ORIGIN.

Tipo de datos: MQCFIN.

Valores:

MQEVO_CONSOLE
Mandato de consola - línea de mandatos.

MQEVO_MSG
Mensaje de mandato del plugin IBM MQ Explorer.

Se devuelve: Siempre.

EventQMgr

Descripción: El gestor de colas en el que se introdujo el mandato o la llamada. (El gestor de colas donde se ejecuta el mandato y que genera el suceso se encuentra en el MD del mensaje de suceso).

Identificador: MQCACF_EVENT_Q_MGR.

Tipo de datos: MQCFST.

Longitud máxima: MQ_Q_MGR_NAME_LENGTH.

Se devuelve: Siempre.

EventAccountingToken

Descripción: Para los mandatos recibidos como mensaje (MQEVO_MSG), es el token contable (AccountingToken) del MD del mensaje de mandato.

Identificador: MQBACF_EVENT_ACCOUNTING_TOKEN.

Tipo de datos: MQCFBS.

Longitud máxima: MQ_ACCOUNTING_TOKEN_LENGTH.

Se devuelve: Sólo si EventOrigin es MQEVO_MSG.

EventIdentityData

Descripción: Para los mandatos recibidos como mensaje (MQEVO_MSG), son los datos de identidad de la aplicación (ApplIdentityData) del MD del mensaje de mandato.

Identificador: MQCACF_EVENT_APPL_IDENTITY.

Tipo de datos: MQCFST.

Longitud máxima: MQ_APPL_IDENTITY_DATA_LENGTH.

Se devuelve: Sólo si EventOrigin es MQEVO_MSG.

EventApplType

Descripción: Para los mandatos recibidos como mensaje (MQEVO_MSG), es el tipo de aplicación (PutApplType) del MD del mensaje de mandato.

Identificador: MQIACF_EVENT_APPL_TYPE.

Tipo de datos: MQCFIN.

Se devuelve: Sólo si EventOrigin es MQEVO_MSG.

EventApplName

Descripción: Para los mandatos recibidos como mensaje (MQEVO_MSG), es el nombre de la aplicación (PutApplName) del MD del mensaje de mandato.

Identificador: MQCACF_EVENT_APPL_NAME.

Tipo de datos: MQCFST.

Longitud máxima: MQ_APPL_NAME_LENGTH.

Se devuelve: Sólo si EventOrigin es MQEVO_MSG.

EventApplOrigin

Descripción: Para los mandatos recibidos como mensaje (MQEVO_MSG), son los datos de origen de la aplicación (ApplOriginData) del MD del mensaje de mandato.

Identificador: MQCACF_EVENT_APPL_ORIGIN.

Tipo de datos: MQCFST.

Longitud máxima: MQ_APPL_ORIGIN_DATA_LENGTH.

Se devuelve: Sólo si EventOrigin es MQEVO_MSG.

Mandato

Descripción: El código del mandato.

Identificador: MQIACF_COMMAND.

Tipo de datos: 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
Estos se definen en IBM MQ 8.0 `cmqcfc.h`

Se devuelve: Siempre.

CommandData contiene elementos PCF que conforman el mandato PCF.

Formato del mensaje de suceso de configuración

Los sucesos de configuración son mensajes PCF de formato Advanced Message Security estándar.

Los valores posibles para el descriptor de mensajes MQMD se pueden encontrar en la sección Mensaje de suceso MQMD (descriptor de mensaje).

Estos son valores de MQMD seleccionados:

```
Format = MQFMT_EVENT
Peristence = MQPER_PERSISTENCE_AS_Q_DEF
PutApplType = MQAT_QMGR           //for both CLI and command server
```

El almacenamiento intermedio de mensaje consta de la estructura MQCFH y la estructura de parámetro que le sigue. Los valores posibles de MQCFH se pueden encontrar en Mensaje de suceso MQCFH (cabecera PCF).

Estos son valores de MQCFH seleccionados:

```
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}
```

Los parámetros que siguen a MQCFH son:

EventUserID

Descripción: El ID de usuario que ha emitido el mandato o la llamada que ha generado el suceso. (Éste es el mismo ID de usuario que se utiliza para comprobar la autorización para poder emitir el mandato o la llamada; para los mandatos recibidos de una cola, éste es también el identificador de usuario (UserIdentifier) del MD del mensaje de mandato).

Identificador: **MQCACF_EVENT_USER_ID**

Tipo de datos: MQCFST.

Longitud máxima: MQ_USER_ID_LENGTH.

Se devuelve: Siempre.

SecurityId

Descripción: Es el valor de MQMD.AccountingToken para el mensaje del servidor de mandatos o Windows SID para el mandato local.

Identificador: **MQBACF_EVENT_SECURITY_ID**

Tipo de datos: MQCBS.

Longitud máxima: MQ_SECURITY_ID_LENGTH.

Se devuelve: Siempre.

EventOrigin

Descripción: El origen de la acción que ha provocado el suceso.

Identificador: **MQIACF_EVENT_ORIGIN**

Tipo de datos: MQCFIN.

Valores: **MQEVO_CONSOLE**
Mandato de consola - línea de mandatos.

MQEVO_MSG
Mensaje de mandato del plugin IBM MQ Explorer.

Se devuelve: Siempre.

EventQMgr

Descripción: El gestor de colas en el que se introdujo el mandato o la llamada. (El gestor de colas donde se ejecuta el mandato y que genera el suceso se encuentra en el MD del mensaje de suceso).

Identificador: **MQCACF_EVENT_Q_MGR**

Tipo de datos: MQCFST

Longitud máxima: MQ_Q_MGR_NAME_LENGTH

Se devuelve: Siempre.

ObjectType

Descripción: Tipo de objeto.

Identificador: **MQIACF_OBJECT_TYPE**

Tipo de datos: MQCFIN

Valor: **MQOT_PROT_POLICY**
Política de protección de Advanced Message Security. **1019** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Se devuelve: Siempre.

PolicyName

Descripción: El nombre de política de Advanced Message Security.

Identificador: **MQCA_POLICY_NAME**.

Tipo de datos: MQCFST.

Valor: **2112** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Longitud máxima: MQ_OBJECT_NAME_LENGTH.

Se devuelve: Siempre.

PolicyVersion

Descripción: Versión de la política de Advanced Message Security.

Identificador: **MQIA_POLICY_VERSION**

Tipo de datos: MQCFIN

Valor **238** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Se devuelve: Siempre

TolerateFlag

Descripción: Distintivo de tolerancia de política de Advanced Message Security.

Identificador: **MQIA_TOLERATE_UNPROTECTED**

Tipo de datos: MQCFIN

Valor **235** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Se devuelve: Siempre.

SignatureAlgorithm

Descripción: Algoritmo de firma de política de Advanced Message Security.

Identificador: **MQIA_SIGNATURE_ALGORITHM**

Tipo de datos: MQCFIN

Valor: **236** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Se devuelve: Siempre que hay un algoritmo de firma definido en la política de Advanced Message Security

EncryptionAlgorithm

Descripción: Algoritmo de cifrado de la política de Advanced Message Security.

Identificador: **MQIA_ENCRYPTION_ALGORITHM**

Tipo de datos: MQCFIN

Valor: **237** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Se devuelve: Siempre que hay un algoritmo de cifrado definido en la política de IBM MQ

SignerDNs

Descripción: Nombre distinguido de los firmantes permitidos.

Identificador: **MQCA_SIGNER_DN**

Tipo de datos: MQCFSL

Valor: **2113** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Longitud máxima: Nombre distinguido de firmante más largo de la política, pero no más largo que MQ_DISTINGUISHED_NAME_LENGTH

Se devuelve: Siempre que está definido en la política de IBM MQ.

RecipientDNs

Descripción: Nombre distinguido de los firmantes permitidos.

Identificador: **MQCA_RECIPIENT_DN**

Tipo de datos: MQCFSL

Valor: **2114** - valor numérico definido en IBM MQ 8.0 o en el archivo `cmqc.h`.

Longitud máxima: Nombre distinguido de destinatario más largo de la política, pero no más largo que MQ_DISTINGUISHED_NAME_LENGTH.

Se devuelve: Siempre que está definido en la política de IBM MQ.

Avisos

Esta información se ha desarrollado para productos y servicios ofrecidos en los Estados Unidos.

Es posible que IBM no ofrezca los productos, servicios o las características que se tratan en este documento en otros países. Consulte al representante local de IBM para obtener información sobre los productos y servicios disponibles actualmente en su zona. Las referencias a programas, productos o servicios de IBM no pretenden indicar ni implicar que sólo puedan utilizarse los productos, programas o servicios de IBM. En su lugar podrá utilizarse cualquier producto, programa o servicio equivalente que no infrinja ninguno de los derechos de propiedad intelectual de IBM. No obstante, es responsabilidad del usuario evaluar y verificar el funcionamiento de cualquier producto, programa o servicio no IBM.

IBM puede tener patentes o solicitudes de patentes pendientes que cubran el tema principal descrito en este documento. El suministro de este documento no le otorga ninguna licencia sobre estas patentes. Puede enviar consultas sobre licencias, por escrito, a:

IBM Director
of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Para consultas sobre licencias relacionadas con información de doble byte (DBCS), póngase en contacto con el Departamento de propiedad intelectual de IBM de su país o envíe las consultas por escrito a:

Licencias de Propiedad Intelectual
Ley de Propiedad intelectual y legal
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokio 103-8510, Japón

El párrafo siguiente no se aplica al Reino Unido ni a ningún otro país donde estas disposiciones contradigan la legislación vigente: INTERNATIONAL BUSINESS MACHINES CORPORATION PROPORCIONA ESTA PUBLICACIÓN "TAL CUAL" SIN NINGÚN TIPO DE GARANTÍA, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUYENDO, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS DE NO INCUMPLIMIENTO, COMERCIABILIDAD O IDONEIDAD PARA UNA FINALIDAD DETERMINADA. Algunas legislaciones no contemplan la exclusión de garantías, ni implícitas ni explícitas, en determinadas transacciones, por lo que puede haber usuarios a los que no les afecte dicha norma.

Esta información puede contener imprecisiones técnicas o errores tipográficos. La información aquí contenida está sometida a cambios periódicos; tales cambios se irán incorporando en nuevas ediciones de la publicación. IBM puede efectuar mejoras y/o cambios en los productos y/o programas descritos en esta publicación en cualquier momento y sin previo aviso.

Cualquier referencia en esta información a sitios web que no son de IBM se realiza por razones prácticas y de ninguna manera sirve como un respaldo de dichos sitios web. Los materiales de dichos sitios web no forman parte de este producto de IBM y la utilización de los mismos será por cuenta y riesgo del usuario.

IBM puede utilizar o distribuir cualquier información que el usuario le proporcione del modo que considere apropiado sin incurrir por ello en ninguna obligación con respeto al usuario.

Los titulares de licencias de este programa que deseen información del mismo con el fin de permitir: (i) el intercambio de información entre los programas creados de forma independiente y otros programas (incluido este) y (ii) el uso mutuo de la información intercambiada, deben ponerse en contacto con:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N

Rochester, MN 55901
U.S.A.

Dicha información puede estar disponible, sujeta a los términos y condiciones apropiados, incluyendo, en algunos casos, el pago de una cantidad.

El programa bajo licencia que se describe en esta información y todo el material bajo licencia disponible para el mismo lo proporciona IBM bajo los términos del Acuerdo de cliente de IBM, el Acuerdo de licencia de programas internacional de IBM o cualquier acuerdo equivalente entre las partes.

Los datos de rendimiento incluidos en este documento se han obtenido en un entorno controlado. Por consiguiente, los resultados obtenidos en otros entornos operativos pueden variar de manera significativa. Es posible que algunas mediciones se hayan realizado en sistemas en nivel de desarrollo y no existe ninguna garantía de que estas mediciones serán las mismas en sistemas disponibles generalmente. Además, algunas mediciones pueden haberse estimado por extrapolación. Los resultados reales pueden variar. Los usuarios de este documento deben verificar los datos aplicables a su entorno específico.

La información relativa a productos que no son de IBM se obtuvo de los proveedores de esos productos, sus anuncios publicados u otras fuentes de disponibilidad pública. IBM no ha comprobado estos productos y no puede confirmar la precisión de su rendimiento, compatibilidad o alguna reclamación relacionada con productos que no sean de IBM. Las preguntas relacionadas con las posibilidades de los productos que no sean de IBM deben dirigirse a los proveedores de dichos productos.

Todas las declaraciones relacionadas con una futura intención o tendencia de IBM están sujetas a cambios o se pueden retirar sin previo aviso y sólo representan metas y objetivos.

Este documento contiene ejemplos de datos e informes que se utilizan diariamente en la actividad de la empresa. Para ilustrar los ejemplos de la forma más completa posible, éstos incluyen nombres de personas, empresas, marcas y productos. Todos estos nombres son ficticios y cualquier similitud con los nombres y direcciones utilizados por una empresa real es puramente casual.

LICENCIA DE COPYRIGHT:

Esta información contiene programas de aplicación de ejemplo en lenguaje fuente que ilustran técnicas de programación en diversas plataformas operativas. Puede copiar, modificar y distribuir estos programas de ejemplo de cualquier forma sin pagar ninguna cuota a IBM para fines de desarrollo, uso, marketing o distribución de programas de aplicación que se ajusten a la interfaz de programación de aplicaciones para la plataforma operativa para la que se han escrito los programas de ejemplo. Los ejemplos no se han probado minuciosamente bajo todas las condiciones. IBM, por tanto, no puede garantizar la fiabilidad, servicio o funciones de estos programas.

Puede que si visualiza esta información en copia software, las fotografías e ilustraciones a color no aparezcan.

Información acerca de las interfaces de programación

La información de interfaz de programación, si se proporciona, está pensada para ayudarle a crear software de aplicación para su uso con este programa.

Este manual contiene información sobre las interfaces de programación previstas que permiten al cliente escribir programas para obtener los servicios de WebSphere MQ.

Sin embargo, esta información puede contener también información de diagnóstico, modificación y ajustes. La información de diagnóstico, modificación y ajustes se proporciona para ayudarle a depurar el software de aplicación.

Importante: No utilice esta información de diagnóstico, modificación y ajuste como interfaz de programación porque está sujeta a cambios.

Marcas registradas

IBM, el logotipo de IBM , ibm.com, son marcas registradas de IBM Corporation, registradas en muchas jurisdicciones de todo el mundo. Hay disponible una lista actual de marcas registradas de IBM en la web en "Copyright and trademark information"www.ibm.com/legal/copytrade.shtml. Otros nombres de productos y servicios pueden ser marcas registradas de IBM o de otras empresas.

Microsoft y Windows son marcas registradas de Microsoft Corporation en EE.UU. y/o en otros países.

UNIX es una marca registrada de Open Group en Estados Unidos y en otros países.

Linux es una marca registrada de Linus Torvalds en Estados Unidos y en otros países.

Este producto incluye software desarrollado por Eclipse Project (http://www.eclipse.org/).

Java y todas las marcas registradas y logotipos son marcas registradas de Oracle o sus afiliados.

Número Pieza:

(1P) P/N: