7.5

Protección de IBM WebSphere MQ

Nota

Antes de utilizar esta información y el producto al que da soporte, lea la información en "Avisos" en la página 333.

Esta edición se aplica a la versión 7 release 5 de IBM® WebSphere MQ y a todos los releases y modificaciones posteriores hasta que se indique lo contrario en nuevas ediciones.

Cuando envía información a IBM, otorga a IBM un derecho no exclusivo para utilizar o distribuir la información de la forma que considere adecuada, sin incurrir por ello en ninguna obligación con el remitente.

Contenido

Seguridad

La seguridad es una consideración importante tanto para desarrolladores de aplicaciones de IBM WebSphere MQ como para administradores del sistema que configuran autorizaciones de IBM WebSphere MQ.

Visión general de la seguridad

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

Los aspectos de seguridad comúnmente aceptados son los siguientes:

- “Identificación y autenticación” en la página 5
- “Autorización” en la página 6
- “Auditoría” en la página 6
- “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 9
- “Infraestructura de claves públicas (PKI)” en la página 14

Cuando planifique una implementación de IBM WebSphere 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 48.

Conceptos relacionados

“Conexión de dos gestores de colas utilizando SSL o TLS” en la página 212
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.

“Trabajar con SSL o TLS” en la página 114
Estos temas proporcionan instrucciones para realizar tareas individuales relacionados con la utilización de SSL o TLS con IBM WebSphere MQ.

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

Conceptos relacionados

“Identificación y autenticación en IBM WebSphere MQ” en la página 22
En IBM WebSphere 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 WebSphere MQ” en la página 22
Puede utilizar la autorización para limitar lo que pueden hacer determinadas personas o aplicaciones en el entorno de IBM WebSphere 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 51 y los subtemas asociados.

Conceptos relacionados

“Auditoría en IBM WebSphere MQ” en la página 22

IBM WebSphere 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 WebSphere MQ.

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

Conceptos relacionados

“Criptografía en IBM WebSphere MQ” en la página 23
IBM WebSphere MQ proporciona cifrado utilizando los protocolos SSL (Secure sockets Layer) y TLS (Transport Security Layer).

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*).

Consulte el Glosario para obtener una definición de cifrado.

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 y TLS” en la página 20 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*.Figura 1 en la página 8 ilustra criptografía de clave simétrica.
- 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*.Figura 2 en la página 8 ilustra la criptografía de clave asimétrica, que también se conoce como *criptografía de clave pública*.

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

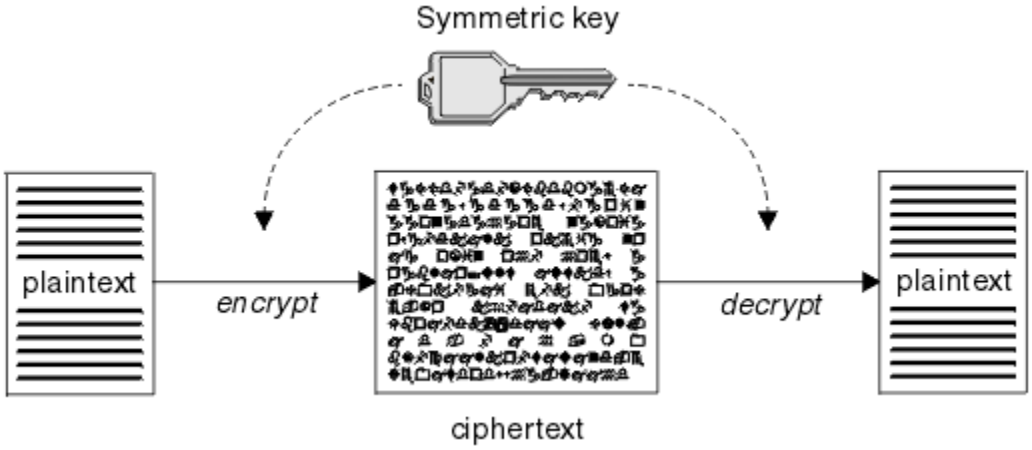


Figura 1. Cifrado de claves simétricas

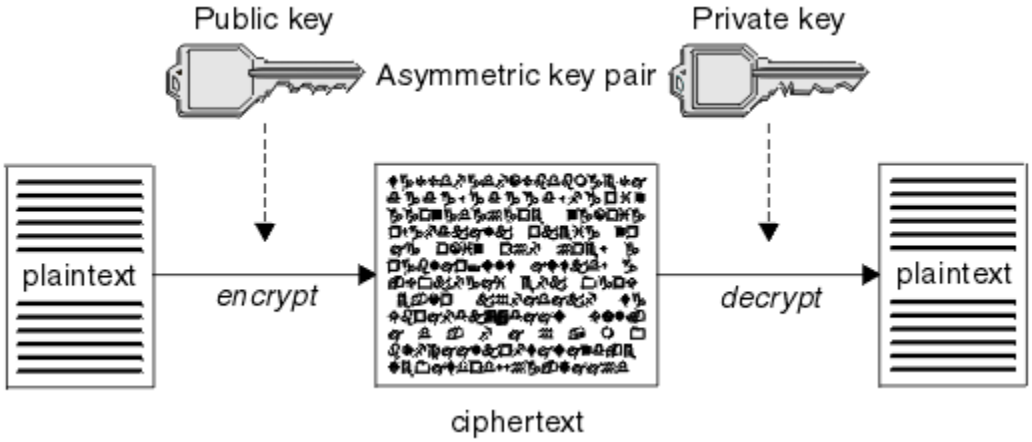


Figura 2. Cifrado de claves asimétricas

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, calculada mediante una función hash. Un resumen de mensaje se puede cifrar, formando una firma digital.

Los mensajes son intrínsecamente variables en tamaño. Un resumen de mensaje es una representación numérica de tamaño fijo del contenido de un mensaje. Un resumen de mensaje se calcula mediante una función hash, que es una transformación que cumple dos criterios:

- La función hash 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 y TLS” en la página 20
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 WebSphere 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 hacer caso omiso de la misma 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
WebSphere MQ da soporte a certificados digitales que cumplan con el estándar X.509. Requiere la opción de autenticación de cliente.

Puesto que IBM WebSphere MQ es un sistema de igual a igual, se considera una autenticación de cliente en la terminología SSL. Por lo tanto, cualquier certificado personal utilizado para la autenticación SSL debe permitir un uso de claves de autenticación del 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 WebSphere MQ, consulte “Políticas de validación de certificados en IBM WebSphere MQ” en la página 35.

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 155.
- 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.

Los tipos de atributos siguientes se encuentran comúnmente en el DN:

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
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 WebSphere MQ tolera ciertos nombres distinguidos (DN) malformados. Para obtener más información, consulte Reglas de WebSphere 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 certificado normalmente se generan mediante la herramienta de gestión de certificados que utiliza el sistema, por ejemplo, la herramienta iKeyman en sistemas UNIX, Linux® y Windows y 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.

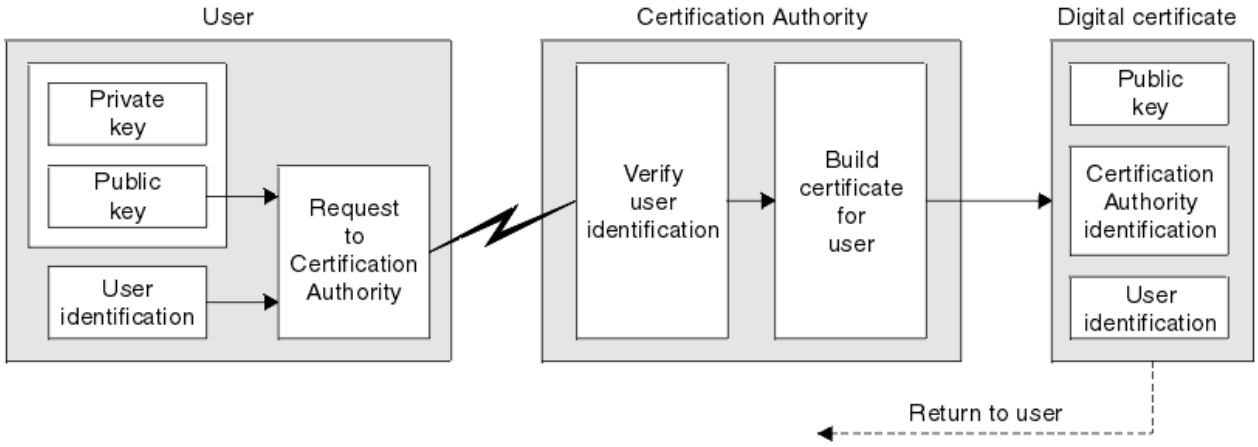


Figura 3. Obtención de un certificado digital

En el diagrama:

- "Identificación de usuario" incluye el nombre distinguido de asunto.
- "Identificación de la entidad emisora de certificados" incluye el nombre distinguido de la CA que emite 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.

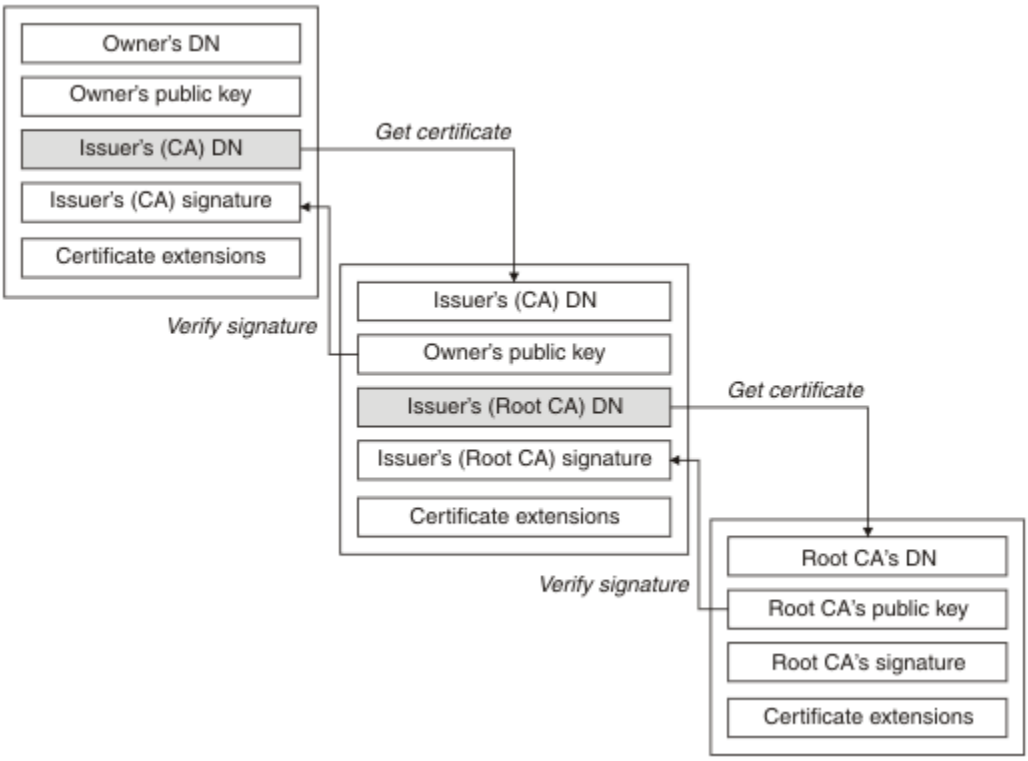


Figura 4. Cadena fiable

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.

Consulte el Glosario para obtener una definición de la caducidad de certificados.

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.

WebSphere MQ puede comprobar si un certificado se ha revocado enviando una solicitud a un respondedor OCSP (protocolo de estado de certificados en línea) (únicamente en los sistemas UNIX, Linux y Windows). O bien, pueden acceder a una CRL 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 155.

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 9 para obtener más información sobre los certificados digitales y las entidades emisoras de certificados (CA). Las RA verifican la información proporcionada 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: SSL y 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 WebSphere MQ da soporte tanto a SSL como a 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 en IBM WebSphere MQ” en la página 24
IBM WebSphere MQ da soporte tanto al protocolo TLS (Transport Layer Security) como SSL (Secure Sockets Layer) para proporcionar seguridad a nivel de enlace para los canales de mensajes y los canales MQI.

Conceptos SSL (Secure Sockets Layer) y TLS (Transport Layer Security)

Los protocolos SSL y TLS permiten 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.

Los protocolos SSL y TLS proporcionan 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 SSL o TLS, que se convierte en el cliente SSL o TLS. La aplicación que recibe la conexión pasa a ser el servidor SSL o TLS. Cada nueva sesión se inicia con un reconocimiento, tal como lo definen los protocolos SSL o TLS.

Se proporciona una lista completa de las CipherSpecs soportadas por IBM WebSphere MQ en “Especificación de CipherSpecs” en la página 223.

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 o TLS

El reconocimiento SSL o TLS permite que el cliente y el servidor SSL o 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 SSL o 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. A continuación, SSL o TLS utiliza la clave compartida para el cifrado simétrico de mensajes, que 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 SSL son los siguientes:

1. El cliente SSL o TLS envía un mensaje de "saludo del cliente" que lista la información de cifrado como, por ejemplo, la versión de SSL o 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 SSL o 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 SSL o TLS verifica el certificado digital del servidor. Para obtener más información, consulte “Cómo SSL y TLS proporcionan la identificación, la autenticación, la confidencialidad y la integridad” en la página 17.
4. El cliente SSL o 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 de mensaje posteriores. La serie de bytes aleatorios se cifra con la clave pública del servidor.
5. Si el servidor SSL o 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 SSL o TLS verifica el certificado del cliente. Para obtener más información, consulte “Cómo SSL y TLS proporcionan la identificación, la autenticación, la confidencialidad y la integridad” en la página 17.
7. El cliente SSL o 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 SSL o 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 SSL o 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 SSL o TLS.

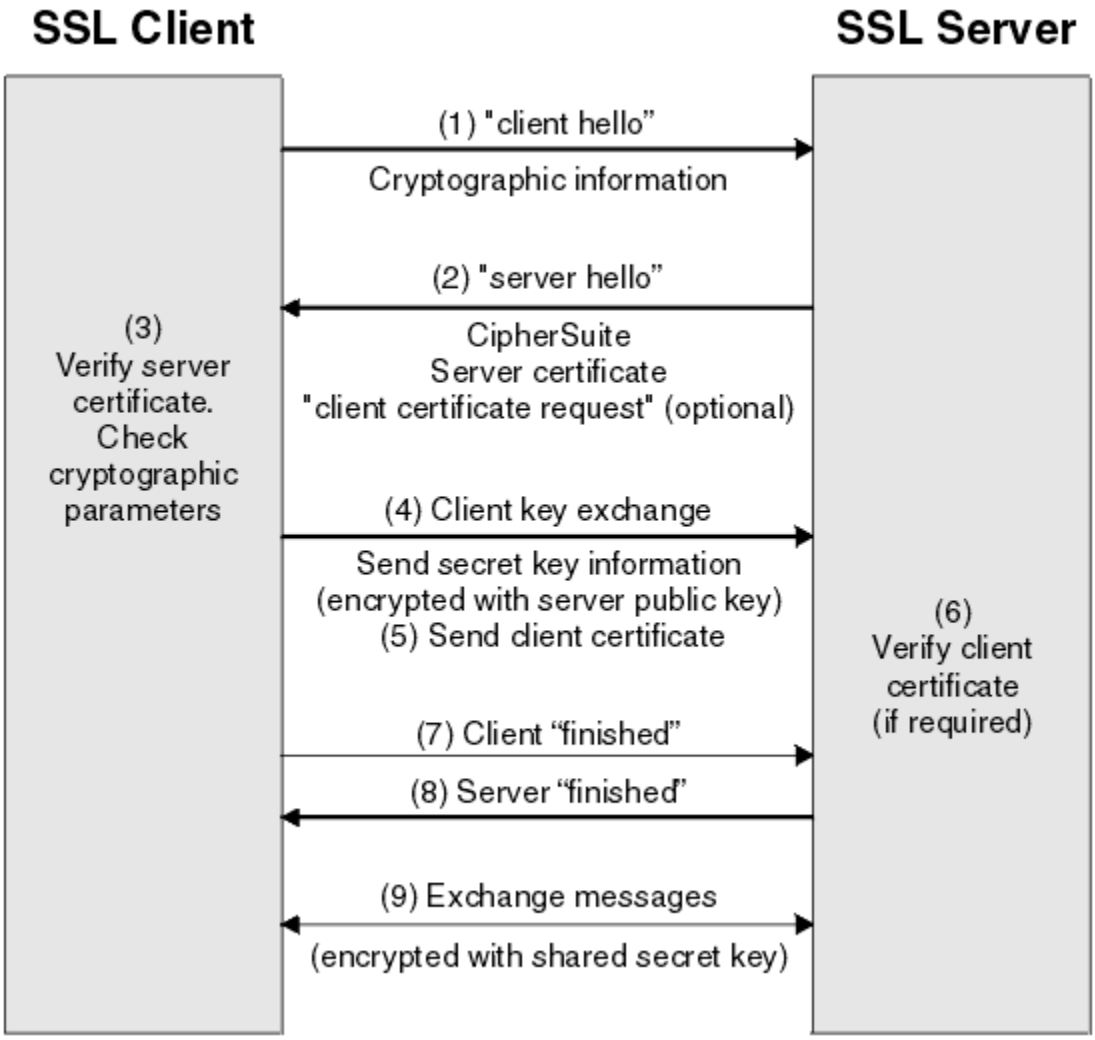


Figura 5. Visión general del reconocimiento SSL o TLS

Cómo SSL y TLS proporcionan la identificación, la autenticación, la confidencialidad y la 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 o TLS" en la página 16.

Cómo SSL y TLS proporcionan 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 SSL o 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 SSL o TLS y la CA Y la que emite el certificado para el servidor SSL o TLS:

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

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

y el cliente SSL o TLS necesita:

- El certificado de CA de la CA Y

Si el servidor SSL o TLS requiere la autenticación del cliente, el servidor comprueba la identidad del cliente verificando el certificado digital del cliente con la clave pública para la CA que ha emitido el certificado personal para el cliente, en este caso la 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 SSL o 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 SSL o TLS verifica el certificado del servidor, y el servidor SSL o TLS verifica el certificado del cliente. Hay cuatro aspectos en esta verificación:

1. La firma digital se comprueba (consulte “Firmas digitales en SSL y TLS” en la página 20).
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 155).

Restablecimiento de claves secretas

Durante un reconocimiento SSL o TLS, se genera una *clave secreta* para cifrar datos entre el cliente y el servidor SSL o 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 SSL y TLS proporcionan confidencialidad

SSL y TLS utilizan una combinación de cifrado simétrico y asimétrico para asegurar la confidencialidad de los mensajes. Durante el reconocimiento SSL o TLS, el cliente y el servidor SSL o 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 SSL o TLS se cifran utilizando este algoritmo y esta clave, lo que garantiza la confidencialidad del mensaje incluso si resulta interceptado. SSL da soporte a una amplia gama de algoritmos de cifrado. Dado que SSL y TLS utilizan 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 SSL y TLS proporcionan integridad

SSL y TLS proporcionan integridad de datos calculando un resumen de mensaje. Para obtener más información, consulte “Integridad de datos de mensajes” en la página 232.

El uso de SSL o 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 “Especificación de CipherSpecs” en la página 223.

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 o SSL deben estar de acuerdo en la misma CipherSpec para poderse comunicar.

Importante: Cuando se trata con canales IBM WebSphere 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 “Especificación de CipherSpecs” en la página 223.

Una Suite de cifrado (CipherSuite) es una suite de algoritmos de cifrado utilizada por una conexión SSL o 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 sólo determinadas combinaciones son válidas cuando se especifican para una conexión TLS o SSL. TLS. El nombre de una CipherSuite válida define la combinación de algoritmos utilizados. Por ejemplo, la CipherSuite SSL_RSA_WITH_RC4_128_MD5 especifica:

- El algoritmo de intercambio y autenticación de claves RSA
- El algoritmo de cifrado RC4, mediante una clave de 128 bits.
- El algoritmo MAC, MD5

Hay varios algoritmos disponibles para el intercambio y autenticación de claves, pero el algoritmo RSA es actualmente el más utilizado. Hay más variedad en los algoritmos de cifrado y algoritmos MAC que se utilizan.

Firmas digitales en SSL y 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.

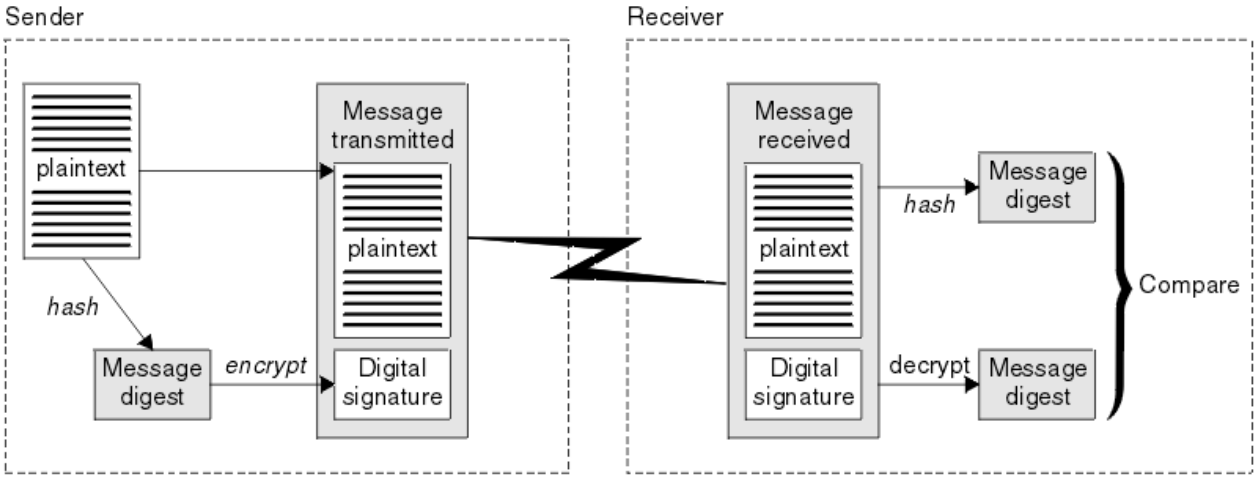


Figura 6. El proceso de firma digital

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 WebSphere 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.

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 versión 1.2. Cuando se configure para el funcionamiento compatible con Suite B, sólo se utilizará el conjunto restringido de algoritmos de cifrado que figura más arriba.
2. Un perfil transitorio para su uso con TLS versión 1.0, o TLS versión 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 Windows, los sistemas UNIX y Linux , WebSphere MQ, se pueden configurar para ajustarse al perfil TLS 1.2 compatible con Suite B, pero no da soporte al perfil transitorio Suite B. Para obtener más información, consulte “NSA Suite B Cryptography en IBM WebSphere MQ” en la página 32.

Información 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).

IBM WebSphere MQ mecanismos de seguridad

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

IBM WebSphere 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 WebSphere MQ

En IBM WebSphere 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 WebSphere 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 5
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 WebSphere MQ

Puede utilizar la autorización para limitar lo que pueden hacer determinadas personas o aplicaciones en el entorno de IBM WebSphere MQ .

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

- Permitir que solo un administrador autorizado emita mandatos para gestionar recursos de IBM WebSphere 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 51 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 WebSphere MQ

IBM WebSphere 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 WebSphere 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 6

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 WebSphere MQ

Puede implementar la confidencialidad en IBM WebSphere MQ cifrando mensajes.

A continuación, se muestran algunos ejemplos de cómo se puede garantizar la confidencialidad en un entorno de IBM WebSphere MQ:

- Después de que un MCA emisor obtenga un mensaje de una cola de transmisión, IBM WebSphere MQ utiliza SSL o 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 WebSphere MQ se podrían considerar suficientes para proteger su contenido contra una revelación no autorizada. Sin embargo, para un mayor nivel de seguridad, puede utilizar IBM WebSphere MQ Advanced Message Security para cifrar los mensajes almacenados en las colas.

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 datos en IBM WebSphere MQ

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

A continuación se muestran algunos ejemplos de cómo se puede garantizar la integridad de los datos en un entorno de IBM WebSphere MQ:

- Puede utilizar SSL o TLS para detectar si el contenido de un mensaje se ha modificado de forma deliberada mientras se transmitía a través de una red. En SSL y TLS, el algoritmo de resumen de mensaje proporciona la detección de mensajes modificados en tránsito. Todas las CipherSpecs de IBM WebSphere MQ proporcionan un algoritmo de resumen de mensaje, excepto para TLS_RSA_WITH_NULL_NULL que no proporcionan integridad de los datos del mensaje.
- Mientras los mensajes se almacenan en una cola local, los mecanismos de control de accesos que proporciona IBM WebSphere MQ pueden considerarse suficientes para impedir la modificación deliberada del contenido de los mensajes. Sin embargo, para un mayor nivel de seguridad, puede utilizar IBM WebSphere MQ 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.

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.

Criptografía en IBM WebSphere MQ

IBM WebSphere MQ proporciona cifrado utilizando los protocolos SSL (Secure sockets Layer) y TLS (Transport Security Layer).

Para más información, consulte “Protocolos de seguridad en IBM WebSphere 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 WebSphere MQ.

Protocolos de seguridad en IBM WebSphere MQ

IBM WebSphere MQ da soporte tanto al protocolo TLS (Transport Layer Security) como SSL (Secure Sockets Layer) 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 SSL o TLS para proporcionar seguridad a nivel de enlace.Un MCA de llamada es un cliente SSL o TLS y un MCA de respuesta es un servidor SSL o TLS. WebSphere MQ da soporte a la versión 3.0 del protocolo SSL y a la versión 1.0 y la versión 1.2 del protocolo TLS (Seguridad de la capa de transporte). Debe especificar los algoritmos de cifrado que utiliza el protocolo SSL o TSL suministrando una CipherSpec como parte de la definición de 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 SSL o TLS, el MCA envía el certificado digital del gestor de colas a su MCA asociado en el otro extremo del canal. El código de WebSphere MQ en el extremo del cliente de un canal MQI actúa en nombre del usuario de la aplicación cliente de WebSphere MQ. Durante el reconocimiento SSL o TLS, el código de WebSphere MQ envía el certificado digital del usuario al MCA en el extremo de servidor del canal MQI.

No es necesario que los gestores de colas y los usuarios del cliente WebSphere MQ tengan asociados certificados digitales personales cuando actúan como clientes SSL o TLS, a menos que se especifique SSLCAUTH (REQUIRED) en el lado 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 repositorio de claves que contiene el certificado digital del gestor de colas de . En un sistema cliente WebSphere MQ , la variable de entorno MQSSLKEYR especifica la ubicación del repositorio de claves que contiene el certificado digital del usuario. De forma alternativa, una aplicación cliente WebSphere MQ puede especificar su ubicación en el campo *KeyRepository* de la estructura de opciones de configuración SSL y TLS, MQSCO, en una llamada MQCONNX. Consulte los temas relacionados para obtener más información sobre los depósitos de claves y cómo especificar su ubicación.

Conceptos relacionados

“Protocolos de seguridad de cifrado: SSL y 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 WebSphere MQ da soporte tanto a SSL como a TLS.

Soporte de IBM WebSphere MQ para SSL y TLS

IBM WebSphere MQ da soporte al protocolo SSL (Secure Sockets Layer) y al protocolo TLS (Transport Layer Security).

Para obtener más información acerca de los protocolos SSL y TLS, consulte la información relacionada.

IBM WebSphere MQ proporciona el siguiente soporte para SSL Versión 3.0 y TLS 1.0 y TLS 1.2:

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

Sistemas UNIX, Linux, and Windowsy HP Integrity NonStop Server
Para sistemas UNIX, Linux, and Windows, y HP Integrity NonStop Server, el soporte SSL y TLS se instala con IBM WebSphere MQ.

Para obtener información sobre los requisitos previos para el soporte SSL y TLS de IBM WebSphere MQ , consulte Requisitos del sistema para IBM WebSphere MQ.

El repositorio de claves SSL o TLS
Una conexión SSL o TLS mutuamente autenticada requiere un repositorio de claves (que puede ser conocido con diferentes nombres en diferentes plataformas) 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. Los nombres de almacén específicos utilizados en las plataformas y los entornos que dan soporte a SSL y TLS son los siguientes:

Java y JMS	almacén de claves y almacén de confianza
Windows UNIX Linux Windows UNIX Linux Sistemas Windows, UNIX and Linux	archivo de base de datos de claves

Para obtener más información, consulte “Certificados digitales” en la página 9 y “Conceptos SSL (Secure Sockets Layer) y TLS (Transport Layer Security)” en la página 15.

Una conexión SSL o TLS mutuamente autenticada requiere un repositorio de claves en cada extremo de la conexión. El repositorio de claves puede contener:

- 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 de certificados con cada gestor de colas o cliente MQI de WebSphere MQ. Los certificados personales son esenciales en un cliente SSL o TLS si se requiere autenticación mutua. 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 repositorio de claves, consulte “Protección de repositorios de claves de IBM WebSphere MQ” en la página 26.

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

Windows UNIX Linux **Windows, sistemas UNIX and Linux**

En Windows, los sistemas UNIX and Linux el repositorio de claves es un archivo 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 WebSphere MQ está instalado en la ubicación predeterminada, la vía de acceso equivalente en Windows es `C:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\key.kdb`.

En sistemas Windows, UNIX and Linux, 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: En Windows, los sistemas UNIX and Linux , las tarjetas de hardware criptográfico PKCS #11 pueden contener los certificados y las claves que, de lo contrario, se conservan en un archivo de base de datos de claves. Cuando los certificados y las claves se guardan en tarjetas PKCS #11, WebSphere MQ continúa necesitando acceso al archivo de base de datos de claves y a un archivo oculto de contraseña.

En los sistemas Windows y UNIX, la base de datos de claves también contiene la clave privada del certificado personal asociado al gestor de colas o al cliente MQI de WebSphere MQ.

Protección de repositorios de claves de IBM WebSphere MQ
El repositorio de claves para IBM WebSphere 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 WebSphere 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.

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 WebSphere MQ Explorer. Para obtener más información, consulte el Mandato MQCMD_REFRESH_SECURITY y el tema *Renovación de la seguridad de SSL o TLS* en la sección de WebSphere MQ Explorer de la documentación de este producto.

Conceptos relacionados
“Renovación de la vista del contenido del repositorio de claves SSL y de los valores SSL de un cliente” en la página 26
Para actualizar la aplicación cliente con el contenido renovado del repositorio de claves, debe detener y reiniciar la aplicación cliente.

Renovación de la vista del contenido del repositorio de claves SSL y de los valores SSL de un cliente
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 de WebSphere 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 26

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).

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 SSL o TLS, para sistemas Windows, UNIX and Linux y z/OS

La conformidad con FIPS 140-2 de una conexión IBM WebSphere MQ SSL o TLS en sistemas UNIX, Linuxy Windows se encuentra aquí "Federal Information Processing Standards (FIPS) para UNIX, Linuxy Windows" en la página 27.

Si el hardware de cifrado está presente, los módulos de cifrado utilizados por IBM WebSphere 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 WebSphere 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 112
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 iKeyman, iKeycmd, runmqakm y runmqckm" en la página 118
En sistemas UNIX, Linux y Windows , gestione claves y certificados digitales con la GUI de iKeyman o desde la línea de mandatos utilizando iKeycmd o runmqakm.

Tareas relacionadas

Habilitar SSL en WebSphere MQ classes for Java

Utilización de SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS

Referencia relacionada

Propiedades SSL de los objetos JMS

Información 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).

Federal Information Processing Standards (FIPS) para UNIX, Linuxy Windows
Cuando la criptografía es necesaria en un canal SSL o TLS en Windows, sistemas UNIX and Linux , WebSphere 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 FIPS (Federal Information Processing Standards) Cryptomodule Validation Program del US National Institute of Standards and Technology, en el nivel 140-2.

La compatibilidad con FIPS 140-2 de una conexión SSL o TLS de WebSphere MQ en los sistemas Windows, UNIX and Linux es la siguiente:

- Para todos los canales de mensajes de IBM WebSphere 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 cliente MQI de IBM WebSphere MQ, 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 clases IBM WebSphere MQ para aplicaciones Java que utilizan la modalidad de cliente, la conexión utiliza las implementaciones SSL y TLS de JRE y es compatible con FIPS si se cumplen las condiciones siguientes:
  - Java Runtime Environment utilizado para ejecutar la aplicación es compatible con FIPS en la versión del sistema operativo y la arquitectura de hardware instalados.
  - Ha especificado que sólo se va a utilizar la criptografía certificada 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 WebSphere MQ classes for JMS que utilizan la modalidad de cliente, la conexión utiliza las implementaciones SSL y TLS de JRE y es compatible con FIPS si se cumplen las condiciones siguientes:
  - Java Runtime Environment utilizado para ejecutar la aplicación es compatible con FIPS en la versión del sistema operativo y la arquitectura de hardware instalados.
  - 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 .NET XMS 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 AIX, Linux, HP-UX, Solaris, Windowsy z/OS soportadas están certificadas con FIPS 140-2 excepto como se indica en el archivo léame incluido con cada fixpack o paquete de renovación.

Para las conexiones SSL y 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 WebSphere 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 SSL y TLS 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 compatibilidad con FIPS 140-2 o bien vuelva a compilar la aplicación cliente de 64 bits, porque el código de 64 bits no está afectado.

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

Cuando WebSphere 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, WebSphere 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 229.

WebSphere MQ genera claves de sesión Triple DES 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 112
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 iKeyman, iKeycmd, runmqakm y runmqckm” en la página 118
En sistemas UNIX, Linux y Windows , gestione claves y certificados digitales con la GUI de iKeyman o desde la línea de mandatos utilizando iKeycmd o runmqakm.

Tareas relacionadas

Habilitar SSL en WebSphere MQ classes for Java

Utilización de SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS

Referencia relacionada

Propiedades SSL de los objetos JMS

Información 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).

SSL y TLS en el cliente MQI de IBM WebSphere MQ

IBM WebSphere MQ ofrece soporte a SSL y TLS en los clientes. Puede adaptar el uso de SSL o TLS de varias maneras.

IBM WebSphere MQ proporciona soporte SSL y TLS para clientes IBM WebSphere MQ MQI en sistemas Windows, UNIX and Linux . Si está utilizando clases IBM WebSphere MQ para Java, consulte Utilización de clases WebSphere MQ para Java y si está utilizando clases IBM WebSphere MQ para JMS, consulte Utilización de clases WebSphere MQ para JMS. El resto de esta sección no se aplica a los entornos Java o JMS.

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

- 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 SSL.

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

Si se efectúan cambios en una máquina cliente en el contenido del repositorio de claves SSL, la ubicación del repositorio de claves SSL, la información de autenticación o los parámetros de hardware de cifrado, debe finalizar todas las conexiones SSL 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 SSL. Se utilizarán los nuevos valores SSL. Estos valores son análogos a los actualizados por el mandato REFRESH SECURITY TYPE(SSL) en los sistemas del gestor de colas.

Cuando el cliente MQI de IBM WebSphere MQ se ejecuta en un sistema Windows, UNIX and Linux con hardware criptográfico, 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 SSL y FIPS está soportado en los clientes MQI de IBM WebSphere MQ. Para obtener más información, consulte “Restablecimiento de claves secretas SSL y TLS” en la página 229 y “Federal Information Processing Standards (FIPS) para UNIX, Linuxy Windows” en la página 27.

Consulte “Configuración de la seguridad del cliente MQI de IBM WebSphere MQ” en la página 111 para obtener más información sobre el soporte SSL de clientes MQI de IBM WebSphere MQ.

Tareas relacionadas

Configuración de un cliente utilizando un archivo de configuración

Especificar que un canal MQI utiliza SSL

Para que un canal MQI utilice SSL, el valor del atributo *SSLCipherSpec*, del canal de conexión de cliente debe ser el nombre de un Ciphercliente IBM WebSphere 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 WebSphere MQ MQI 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 WebSphere 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 SSL.

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 cliente MQI de IBM WebSphere MQ" en la página 229
Tiene tres opciones para especificar una CipherSpec para un cliente MQI de IBM WebSphere MQ .

CipherSpecs y CipherSuites en IBM WebSphere MQ

IBM WebSphere MQ da soporte a las CipherSpecs SSL y TLS CipherSpecs y los algoritmos RSA y Diffie-Hellman.

WebSphere MQ da soporte a SSL V3 y TLS V1.0 y V1.2 CipherSpecs.

WebSphere 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 SSL 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 WebSphere MQ
Este tema proporciona información sobre cómo configurar IBM WebSphere MQ en sistemas Windows, Linuxy UNIX para que se ajusten 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 WebSphere 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 WebSphere MQ Version 7.5 muestra la versión de Suite B implementada por cada nivel de mantenimiento de producto. Si configura IBM WebSphere MQ para implantar la conformidad con Suite B, consulte siempre el archivo readme cuando planifique el mantenimiento (consulte Readmes de los productos IBM MQ, WebSphere MQ y MQSeries).

En sistemas Windows, UNIXy Linux , IBM WebSphere MQ se puede configurar para que se ajuste al perfil TLS 1.2 compatible con Suite B en los niveles de seguridad que se muestran en la Tabla 1.

Tabla 1. 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 WebSphere MQ es no cumplir con el estándar Suite B, IBM WebSphere MQ se puede configurar para que se ajuste a uno o ambos niveles de seguridad en los sistemas Windows, UNIX y Linux . Tras la configuración satisfactoria de IBM WebSphere 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 las CipherSpecs de WebSphere MQ, consulte “Especificación de CipherSpecs” en la página 223

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 WebSphere 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 las Opciones runmqckm y runmqakm.

Nota: Las herramientas **iKeycmd** e **iKeyman** no dan soporte a la creación de certificados digitales para que funcionen de forma 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 los sistemas UNIX, Linux, and Windows” en la página 126.

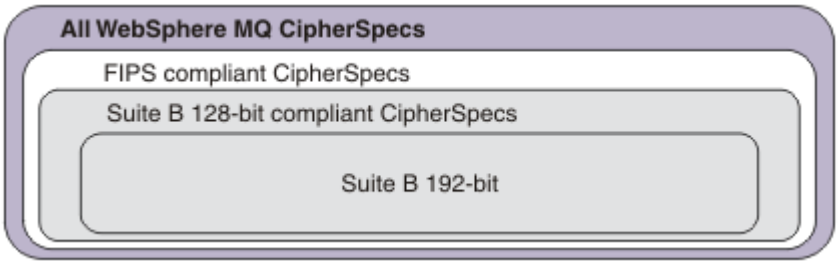
Para solicitar un certificado digital firmado por una CA para su utilización en la producción de Suite B, consulte “Solicitud de un certificado personal en los sistemas UNIX, Linux, and Windows” en la página 128.

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 WebSphere MQ se ha configurado para que tenga un funcionamiento compatible con 140-2. Por consiguiente, es posible configurar WebSphere 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 WebSphere MQ para que sea compatible con Suite B

Para obtener información sobre cómo configurar IBM WebSphere MQ en Windows, UNIX y Linux para el funcionamiento compatible con Suite B, consulte “Configuración de IBM WebSphere MQ para Suite B” en la página 34.

IBM WebSphere MQ no da soporte al funcionamiento de forma compatible con Suite B en las plataformas IBM i y z/OS. Los clientes Java y JMS de WebSphere MQ no 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 112
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 WebSphere MQ para Suite B
IBM WebSphere MQ se puede configurar para que funcione en conformidad con el estándar Suite B de la NSA en los sistemas UNIX, Linux, and Windows.

Suite B restringe el conjunto de algoritmos de cifrado permitidos para proporcionar un nivel de seguridad garantizado.IBM WebSphere MQ se puede configurar para que funcione en conformidad con Suite B para proporcionar un nivel mejorado de seguridad. 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 SSL y TLS, consulte "NSA Suite B Cryptography en IBM WebSphere MQ" en la página 32.

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.

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. Al establecer 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 el uso de Suite B en el archivo de configuración de cliente, consulte Stanza SSL del archivo de configuración de cliente .

Para obtener más información sobre el uso de la variable de entorno MQSUITEB, consulte 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 WebSphere MQ classes for .NET, consulte Clase .NET MQEnvironment.

Políticas de validación de certificados en IBM WebSphere 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 las 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 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 sistemas UNIX, Linux, and Windows, consulte “Configuración de políticas de validación de certificados en IBM WebSphere MQ” en la página 35. Para obtener más información sobre las diferencias entre las políticas de validación de certificados básica y estándar, consulte Validación de certificados y diseño de políticas de confianza en sistemas UNIX, Linux y Windows .

Configuración de políticas de validación de certificados en IBM WebSphere MQ
Puede especificar qué política de validación de certificado SSL/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*. Para obtener más información sobre cómo establecer 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*. Para obtener más información sobre cómo utilizar esta variable, consulte MQCERTVPOL .
3. Utilizando el parámetro de ajuste de la stanza SSL del cliente, *CertificateValPolicy*. Para obtener 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 WebSphere MQ” en la página 35.

Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere 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 WebSphere MQ.

En los releases anteriores de IBM WebSphere MQ, todas las CipherSpecs de SSL y TLS soportadas utilizaban el algoritmo RSA para firmas digitales y acuerdos de claves. Todos los tipos de certificados digitales soportados eran compatibles con todas las CipherSpecs soportadas, de modo que era posible cambiar la CipherSpec de cualquier canal sin necesidad de cambiar los certificados digitales.

En IBM WebSphere MQ v7.5, 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. Si desea 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 SSL 3.0 y 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 `cert_label` es la etiqueta de certificado del algoritmo de firma digital que necesita visualizar.

Nota: Aunque la herramienta **iKeycmd (runmqckm)** y la GUI de **iKeyman (strmqikm)** 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** generará una salida en la que se muestra el uso del algoritmo de firma especificado:

```
Label : ibmwebspheremqexample
Key Size : 1024
Version : X509 V3
Serial : 4e4e93f1
Issuer : CN=Old Certificate Authority,OU=Test,O=Example,C=US
Subject : CN=Example Queue Manager,OU=Test,O=Example,C=US
Not Before : August 19, 2011 5:48:49 PM GMT+01:00
Not After : August 18, 2012 5:48:49 PM GMT+01:00
Public Key
    30 81 9F 30 0D 06 09 2A 86 48 86 F7 0D 01 01 01
    05 00 03 81 8D 00 30 81 89 02 81 81 00 98 5A 7A
    F0 18 21 EE E4 8A 6E DE C8 01 4B 3A 1E 41 90 3D
    CE 01 3F E6 32 30 6C 23 59 F0 FE 78 6D C2 80 EF
    BC 83 54 7A EB 60 80 62 6B F1 52 FE 51 9D C1 61
    80 A5 1C D4 F0 76 C7 15 6D 1F 0D 4D 31 3E DC C6
    A9 20 84 6E 14 A1 46 7D 4C F5 79 4D 37 54 0A 3B
    A9 74 ED E7 8B 0F 80 31 63 1A 0B 20 A5 99 EE 0A
    30 A6 B6 8F 03 97 F6 99 DB 6A 58 89 7F 27 34 DE
```

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

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

No se puede utilizar todas las CipherSpecs con todos los certificados digitales. Existen tres tipos de CipherSpec, que se distinguen mediante el prefijo del nombre de CipherSpec. Cada tipo de CipherSpec impone restricciones diferentes sobre el tipo de certificado digital que se puede utilizar. Estas restricciones se aplican a todas las conexiones SSL y TLS de WebSphere MQ, pero resultan especialmente relevantes para los usuarios del cifrado Elliptic Curve.

En la tabla siguiente se resumen las relaciones entre las CipherSpecs y los certificados digitales:

Tabla 2. 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 Elliptic Curve y algoritmos de firma digital Elliptic Curve.	RSA	RSA	ECDHE

Tabla 2. Relaciones entre las CipherSpecs y los certificados digitales (continuación)

Tipo	Prefijo de nombre de CipherSpec	Descripción	Tipo de clave pública necesaria	Algoritmo de cifrado de firma digital	Método de establecimiento de claves secretas
3	(Todos los demás)	CipherSpecs que utilizan claves públicas RSA y algoritmos de firma digital RSA.	RSA	RSA	RSA

Nota: Los gestores de colas WebSphere MQ y los clientes MQI sólo dan soporte a las CipherSpecs de Tipo 1 y 2 en las plataformas UNIX, Linux, and Windows.

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.

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. Hay varios algoritmos de firma digital que pueden utilizarse, 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 WebSphere 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.

Un gestor de colas de WebSphere MQ sólo puede utilizar un único certificado personal para identificarse. Esto significa que todos los canales del gestor de colas utilizará el mismo certificado digital y, por tanto, cada gestor de colas sólo puede utilizar un tipo de CipherSpec a la vez. De forma parecida, una aplicación cliente de WebSphere MQ sólo puede utilizar un único certificado personal para identificarse. Esto significa que todas las conexiones SSL y TLS de un único proceso de aplicación utilizarán el mismo certificado digital y por consiguiente, cada proceso de aplicación de cliente sólo puede utilizar un tipo de CipherSpec a la vez.

Los tres tipos de CipherSpec no interactúan directamente: se trata de una limitación de los estándares actuales de SSL y TLS. Por ejemplo, supongamos que ha elegido utilizar la CipherSpec ECDHE_ECDSA_AES_128_CBC_SHA256 para un canal receptor llamado TO.QM1 en un gestor de colas llamado QM1. ECDHE_ECDSA_AES_128_CBC_SHA256 es una CipherSpec de Tipo 1, por lo que QM1 debe tener un certificado personal con una clave Elliptic Curve y una firma digital basada en ECDSA. Todos los clientes y otros gestores de colas que se comuniquen directamente con QM1 deben, por tanto, tener los certificados digitales que satisfagan los requisitos de las CipherSpecs de Tipo 1. Los

demás canales que se conecten al gestor de colas QM1 pueden utilizar otras CipherSpecs (por ejemplo ECDHE_ECDSA_3DES_EDE_CBC_SHA256), pero sólo pueden utilizar las CipherSpecs de Tipo 1 para comunicarse con QM1.

Cuando planifique las redes de WebSphere MQ, considere detenidamente qué canales requieren SSL o TLS y asegúrese de que todos los clientes y los gestores de colas que necesiten interoperar utilicen el mismo tipo de CipherSpecs y los certificados digitales apropiados. Los estándares IETF RFC 4492, RFC 5246 y RFC 6460 describen el uso detallado de las CipherSpecs Elliptic Curve en TLS 1.2.

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 `cert_label` es la etiqueta del certificado cuyo algoritmo de firma digital necesita visualizar.

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

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

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 la herramienta **iKeycmd (runmqckm)** con los mismos parámetros. Asimismo, también puede utilizar la GUI de **iKeyman (strmqikm)** para ver los algoritmos de firma digital si abre el depósito de claves y efectúa una doble pulsación en la etiqueta del certificado. No obstante, se recomienda utilizar la herramienta **runmqakm** para ver los certificados digitales, porque da soporte a una gama de algoritmos más amplia.

CipherSpecs Elliptic Curve y Suite B de la NSA

Cuando se configura WebSphere 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 WebSphere MQ” en la página 32. 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 3. 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

“Especificación de CipherSpecs” en la página 223
Especifique una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC de **DEFINE CHANNEL** o en el mandato MQSC de **ALTER CHANNEL** .

“Especificación de que sólo se utilizan CipherSpecs certificadas por FIPS en el tiempo de ejecución del cliente MQI” en la página 112
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 WebSphere MQ” en la página 32
Este tema proporciona información sobre cómo configurar IBM WebSphere MQ en sistemas Windows, Linuxy UNIX para que se ajusten 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.

Valores de CipherSpec soportados en IBM WebSphere MQ

El conjunto de CipherSpecs predeterminadas solamente permite los siguientes valores:

TLS 1.0

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA

TLS 1.2

- ECDHE_ECDSA_AES_128_CBC_SHA256
- ECDHE_ECDSA_AES_256_CBC_SHA384
- ECDHE_ECDSA_AES_128_GCM_SHA256
- ECDHE_ECDSA_AES_256_GCM_SHA384
- ECDHE_RSA_AES_128_CBC_SHA256

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

Habilitación de CipherSpecs en desuso

De forma predeterminada, no está permitido especificar una CipherSpec en desuso en una definición de canal. Si intenta especificar un CipherSpec en desuso, recibirá un mensaje AMQ9788 en el registro de errores del gestor de colas.

Es posible volver a habilitar los CipherSpecs en desuso editando el archivo `qm.ini`. En la stanza SSL del archivo `qm.ini`, añada la siguiente línea:

```
SSL:
AllowWeakCipherSpec=Yes
```

También puede volver a habilitar una o más de las CipherSpecs en desuso durante el tiempo de ejecución en el servidor estableciendo la variable de entorno *AMQ_SSL_WEAK_CIPHER_ENABLE* en cualquier valor. Esta variable de entorno habilita las CipherSpecs independientemente del valor que se especifique en el archivo `qm.ini`.

Registros de autenticación de canal

Para ejercer un control más preciso sobre el acceso que se otorga a la conexión de los sistemas en un nivel de canal, puede utilizar los 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 WebSphere 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 4. 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. Sin embargo, puede haber intentos de conexión desde una dirección IP que no tenga autorización para acceder al sistema de WebSphere MQ. Por tanto, deberá bloquear temporalmente esa dirección hasta que se actualice el cortafuegos. Es posible incluso que estos intentos de conexión no procedan de canales de WebSphere MQ sino de otras aplicaciones de socket que están mal configuradas para orientarse al escucha de WebSphere 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 187 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 189 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 para un ID de usuario confirmado por el cliente" en la página 192 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 191 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 "Bloqueo del acceso para un Nombre distinguido SSL" en la página 192 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 193 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 189 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 WebSphere 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 confirmado por el cliente con un ID de usuario MCAUSER” en la página 190 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 191 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 189 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, dirección IP, nombre de gestor de colas o nombre distinguido de SSL o TLS, se utiliza la coincidencia más específica. La coincidencia más específica se determina de la manera siguiente.

- Para un nombre de canal:
  - La coincidencia más específica es un nombre sin comodines, por ejemplo, A.B.C.
  - La coincidencia más genérica es un asterisco individual (*), que coincide con todos los nombres de canal.
  - Un patrón con un asterisco en la posición más hacia la izquierda es más genérico que un patrón con un valor definido en la posición más hacia la izquierda. Por tanto, *.B.C es más genérico que A.*.
  - Un patrón con un asterisco en la segunda posición es más genérico que un patrón con un valor definido en la segunda posición, y lo mismo ocurre para cada posición subsiguiente. Por tanto, A.*.C es más genérico que A.B.*
  - Cuando dos o más patrones tienen un asterisco en la misma posición, el patrón con menos nodos a continuación del asterisco es más genérico. Así, A. es más genérico que A.*.C
- Para una dirección IP:
  - La coincidencia más específica es un nombre sin comodines, por ejemplo, 192.0.2.6.
  - La coincidencia más genérica es un asterisco individual (*), que coincide con todos los nombres de canal.
  - Un patrón con un asterisco en la posición más hacia la izquierda es más genérico que un patrón con un valor definido en la posición más hacia la izquierda. Por tanto, *.0.2.6 es más genérico que 192.*.

- Un patrón con un asterisco en la segunda posición es más genérico que un patrón con un valor definido en la segunda posición, y lo mismo ocurre para cada posición subsiguiente. Por tanto, 192.*.2.6 es más genérico que 192.0.*.
- Cuando dos o más patrones tienen un asterisco en la misma posición, el patrón con menos nodos a continuación del asterisco es más genérico. Por lo tanto 192 .* es más genérico que 192.*.2.*.
- 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 siempre que no se proporcione ningún asterisco junto al asterisco final. Por ejemplo, 192.*.2.* es válido, pero 192.0.*.* no es válido.
- Un patrón de dirección IPv6 no puede contener dos puntos seguidos y un asterisco final porque la dirección resultante sería ambigua. Por ejemplo, 2001::* podría expandirse a 2001:0000:*, 2001:0000:0000:* etc.

- Para un nombre de gestor de colas:
  - La coincidencia más específica es un nombre sin comodines, por ejemplo, 192.0.2.6.
  - La coincidencia más genérica es un asterisco individual (*), que coincide con todos los nombres de canal.
  - Un patrón con un asterisco en la posición más hacia la izquierda es más genérico que un patrón con un valor definido en la posición más hacia la izquierda. Por tanto, *QUEUEMANAGER es más genérico que QUEUEMANAGER*.
  - Un patrón con un asterisco en la segunda posición es más genérico que un patrón con un valor definido en la segunda posición, y lo mismo ocurre para cada posición subsiguiente. Por tanto, Q*MANAGER es más genérico que QUEUE*.
  - Cuando dos o más patrones tienen un asterisco en la misma posición, el patrón con menos caracteres a continuación del asterisco es más genérico. Por tanto, Q* es más genérico que Q*MGR.
- Para un nombre distinguido de SSL o TLS, el orden de prioridad de las subseries de caracteres es el siguiente:

Tabla 5. Orden de prioridad de subseries

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

Tabla 5. Orden de prioridad de subseries (continuación)		
Orden	**Subserie de nombre distinguido**	**Nombre**
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 declara con un nombre distinguido que contiene las subseries O=IBM y C=UK, WebSphere MQ utiliza un registro de autenticación de canal para O=IBM en preferencia a 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 58
En este apartado se tratan aspectos relativos a la seguridad de la mensajería remota.

Seguridad de mensajes en IBM WebSphere MQ

La seguridad de mensajes de la infraestructura de IBM WebSphere MQ la proporciona un componente con licencia por separado IBM WebSphere MQ Advanced Message Security.

IBM WebSphere MQ Advanced Message Security (AMS) amplía los servicios de seguridad de IBM WebSphere 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

"IBM WebSphere MQ Advanced Message Security" en la página 275
IBM WebSphere MQ Advanced Message Security (AMS) es un componente con licencia separada de IBM WebSphere MQ Advanced Message Security que proporciona un nivel alto de protección para los datos sensibles que fluyen por la red IBM WebSphere MQ Advanced Message Security, 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 WebSphere MQ.

Puede utilizar IBM WebSphere 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.

WebSphere MQ proporciona un rango de servicios de seguridad a nivel de enlace, incluido el soporte para SSL (Secure Sockets Layer) y TLS (Transport Layer Security).

Debe tener en cuenta determinados aspectos de la seguridad cuando implemente WebSphere. En los sistemas UNIX, Linux y Windows, si ignora estas cuestiones y no hace nada, podrá utilizar WebSphere MQ.

Las consideraciones sobre seguridad se describen más abajo.

Autorización para administrar WebSphere MQ

Los administradores de WebSphere MQ necesitan autorización para:

- Emitir mandatos para administrar WebSphere MQ
- Utilizar IBM WebSphere MQ Explorer

Si desea ver más información, consulte:

- "Autorización para administrar IBM WebSphere MQ en sistemas UNIX, Linux, and Windows" en la página 202

Autorización para trabajar con objetos WebSphere MQ

Las aplicaciones pueden acceder a los objetos WebSphere 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 WebSphere MQ y para acceder también a canales y a objetos de información de autenticación. Estos objetos pueden ser protegidos por WebSphere 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 WebSphere MQ" en la página 52.

Seguridad de canal

Los ID de usuario asociados a los agentes de canal de mensajes (MCA) necesitan autorización para acceder a diferentes recursos WebSphere 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 72.

Consideraciones adicionales

Debe tener en cuenta los siguientes aspectos de seguridad sólo si utiliza ciertas extensiones de funciones o de producto base de WebSphere MQ:

- "Seguridad para clústeres de gestores de colas" en la página 81
- "Seguridad para publicación/suscripción de IBM WebSphere MQ" en la página 82
- "Seguridad para IBM WebSphere MQ Internet Pass-thru" en la página 84

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 WebSphere 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 WebSphere MQ que utilizan SSL o 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 WebSphere MQ. Para obtener más información, consulte:

- "Correlacionar un gestor de colas remoto con un ID de usuario MCAUSER" en la página 189
- "Correlación de un ID de usuario confirmado por el cliente con un ID de usuario MCAUSER" en la página 190
- "Correlacionar un Nombre distinguido SSL o TLS con un ID de usuario MCAUSER" en la página 191
- "Correlacionar una dirección IP con un ID de usuario MCAUSER" en la página 193

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 cliente MQI de IBM WebSphere MQ 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.

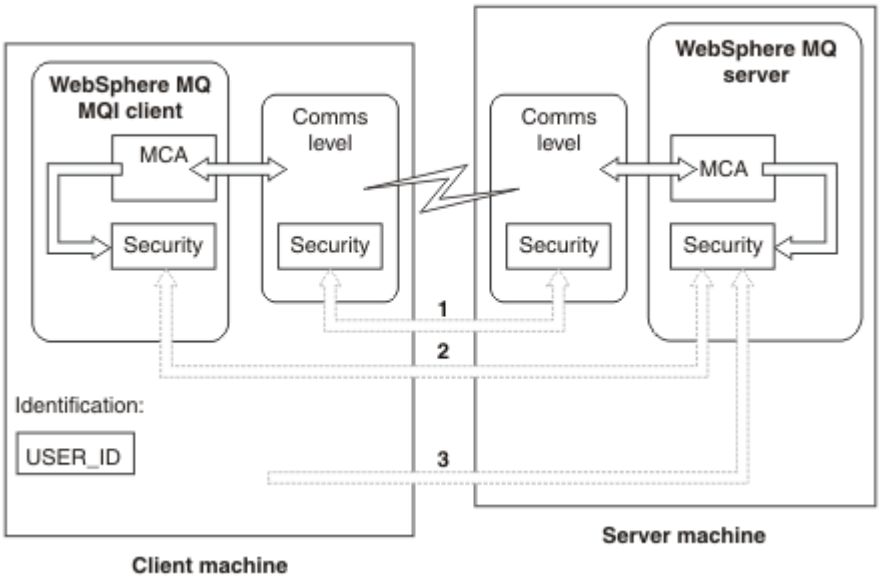


Figura 7. Seguridad en una conexión cliente/servidor

1. Nivel de comunicaciones

   Consulte la flecha 1. Para implementar la seguridad a nivel de comunicaciones, utilice SSL o TLS. Para obtener más información, consulte “Protocolos de seguridad de cifrado: SSL y 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 SSL/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 41 se proporciona una descripción completa.

3. 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.

4. 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 WebSphere 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.

Para ayudar a la identificación en el cliente de IBM WebSphere MQ para HP Integrity NonStop Server, el cliente pasa el alias OSS Safeguard bajo el cual se ejecuta la aplicación cliente. Este ID suele tener el formato <PRIMARYGROUP>.<ALIAS>. Si es necesario, puede correlacionar este ID de usuario con un ID de usuario alternativo en el gestor de colas utilizando los registros de autenticación de canal o una salida de seguridad. Para obtener más información acerca de las salidas de mensajes, consulte la publicación "Correlación de identidad en salidas de mensajes" en la página 153. Si desea más información sobre cómo definir registros de autenticación de canal, consulte "Correlación de un ID de usuario confirmado por el cliente con un ID de usuario MCAUSER" en la página 190.

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 cliente MQI de IBM WebSphere MQ.

ID de usuario

Si el cliente MQI de IBM WebSphere MQ está en Windows y el servidor de IBM WebSphere MQ también está en Windows y tiene acceso al dominio en el que está definido el ID de usuario de cliente, IBM WebSphere MQ da soporte a ID de usuario de hasta 20 caracteres. En las plataformas y configuraciones UNIX and Linux, la longitud máxima es de 12 caracteres.

Un servidor WebSphere MQ para 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`.

Tenga en cuenta que `UNKNOWN` es un ID de usuario reservado y el ID de usuario `NOBODY` también tiene significados especiales para WebSphere MQ. La creación de ID de usuario en el sistema operativo llamado `UNKNOWN` o `NOBODY` podría tener resultados imprevistos.

Aunque los ID de usuario se utilizan para autenticar, los grupos se utilizan para la autorización, excepto para Windows.

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.

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 WebSphere MQ, incluidos los que se conectan desde un cliente MQI de IBM WebSphere MQ.

Para poder utilizar IBM WebSphere 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 WebSphere MQ puede subdividirse en dos categorías principales:

- Autorización para realizar operaciones administrativas
- Autorización para que las aplicaciones utilicen IBM WebSphere 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 WebSphere MQ

Los administradores de IBM WebSphere MQ necesitan autorización para realizar diversas funciones. Esta autorización se obtiene de diferentes maneras en diferentes plataformas.

Los administradores de IBM WebSphere MQ necesitan autorización para:

- Emitir mandatos para administrar IBM WebSphere MQ

- Utilizar IBM WebSphere MQ Explorer

Para obtener más información, consulte el tema correspondiente a su sistema operativo.

Autorización para administrar IBM WebSphere MQ en sistemas UNIX y Windows

Un administrador de IBM WebSphere MQ es un miembro del grupo mqm. Este grupo tiene acceso a todos los recursos de IBM WebSphere MQ y puede emitir mandatos de control IBM WebSphere MQ. Un administrador puede otorgar autorizaciones específicas a otros usuarios.

Para ser un administrador de IBM WebSphere MQ en los sistemas UNIX y Windows, un usuario debe ser miembro del *grupo mqm*. Este grupo se crea automáticamente cuando instala WebSphere MQ. Para permitir que los usuarios emitan mandatos de control, debe añadirlos al grupo mqm. Esto incluye el usuario root en los sistemas 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 WebSphere MQ, y tienen autorización para ejecutar solamente los mandatos para los que se les ha otorgado acceso.

Además, en sistemas Windows, las cuentas SYSTEM y de administrador tienen acceso completo a los recursos de IBM WebSphere MQ.

Todos los miembros del grupo mqm tienen acceso a todos los recursos WebSphere 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 WebSphere MQ.

Los administradores pueden utilizar el mandato **runmqsc** para emitir mandatos de script de WebSphere 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.

WebSphere MQ Explorer emite mandatos PCF para realizar tareas de administración. No es necesario que los administradores tengan autorizaciones adicionales si desean utilizar WebSphere MQ Explorer para administrar un gestor de colas en el sistema local. Cuando se utiliza WebSphere MQ Explorer para administrar un gestor de colas en otro sistema, los administradores deben tener las autorizaciones necesarias para que los gestores de colas remotos procesen 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 WebSphere MQ” en la página 52.
- Para los mandatos que se ejecutan en canales, iniciadores de canal, escuchas y clústeres, consulte Seguridad de canal.

Para obtener más información sobre la autorización que necesita para administrar WebSphere MQ En los sistemas UNIX y Windows, consulte la información relacionada.

Autorización para que las aplicaciones utilicen IBM WebSphere 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 WebSphere 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 WebSphere 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.

Las aplicaciones que utilizan clases IBM WebSphere MQ para Java, clases IBM WebSphere MQ para JMS, clases IBM WebSphere MQ para .NET o Message Service Clients for C/C++ and .NET utilizan MQI indirectamente.

Los MCA también emiten llamadas MQI y los ID de usuario asociados a los MCA necesitan autorización para acceder a estos objetos WebSphere MQ. Para obtener más información acerca de estos ID de usuario y las autorizaciones que necesitan, consulte "Autorización de canal" en la página 72.

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.

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 WebSphere MQ. IBM WebSphere 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 WebSphere 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 UNIX, Linux, and Windows, 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 `NombreObjeto` del descriptor 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 WebSphere
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 54.

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.

Autorización para trabajar con objetos IBM WebSphere MQ en sistemas UNIX, Linux y Windows

El componente de servicio de autorización suministrado con IBM WebSphere MQ se denomina *OAM (gestor de autorizaciones sobre objetos)*. Proporciona control de acceso a través de comprobaciones de autenticación y autorización.

1. AUTENTICACIÓN.

   La comprobación de la autenticación ejecutada por el OAM que se suministra con IBM WebSphere 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 la comprobación de autenticación cuando una aplicación se conecta a un gestor de colas, y se cumplen las siguientes condiciones.

   Si la estructura MQCSP la ha proporcionado la aplicación de conexión, y al atributo *AuthenticationType* de la estructura MQCSP se le asigna el valor MQCSP_AUTH_USER_ID_AND_PWD, la comprobación la realiza el OAM en su función MQZID_AUTHENTICATE_USER. Esta es la comprobación: el ID de usuario de la estructura MQCSP se compara con el ID de usuario de *IdentityContext* (MQZIC), para determinar si coinciden. Si no coinciden, la comprobación no se realiza correctamente.

   Esta comprobación básica no está pensada para ser una autenticación completa del usuario. Por ejemplo, no hay ninguna comprobación de la autenticidad del usuario mediante la comprobación de la contraseña suministrada en la estructura MQCSP. Asimismo, si la aplicación omite una estructura MQCSP, entonces no se realiza la comprobación.

   Si son necesarios servicios de autenticación más completos a través del componente de servicio de autorización, el OAM proporcionado con IBM WebSphere MQ no ofrece esta posibilidad. Debe escribir un nuevo componente de servicio de autorización o bien obtener uno de un proveedor.

2. 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 sistemas UNIX, Linux y Windows, el *servicio de autorización* proporciona el control de accesos cuando una aplicación emite una llamada MQI para acceder a un objeto de IBM WebSphere 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.

En Windows, el OAM otorga a los miembros del grupo Administradores la autorización para acceder a todos los objetos de IBM WebSphere MQ, aunque el UAC esté habilitado.

Además, en sistemas Windows , la cuenta SYSTEM tiene acceso completo a los recursos de IBM WebSphere MQ .

El servicio de autorización también proporciona comprobaciones de autorización cuando un mandato PCF realiza operaciones en uno de estos objetos de IBM WebSphere 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.

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 WebSphere MQ.

El componente de servicio de autorización suministrado con IBM WebSphere MQ se denomina *OAM (gestor de autorizaciones sobre objetos)*. 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 WebSphere 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 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 aumentado este límite, pero IBM WebSphere MQ sigue observando una restricción de 12 caracteres en todas las plataformas UNIX . Si utiliza un ID de usuario de más de 12 caracteres, IBM WebSphere 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 WebSphere 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.

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 a los miembros del grupo VOYAGER colocar mensajes en cualquier cola con un nombre que 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.

Si no desea realizar comprobaciones de seguridad como sería el caso, por ejemplo, de un entorno de prueba, puede inhabilitar el OAM.

Utilización de PCF para acceder a los mandatos del gestor de autorizaciones sobre objetos (OAM)
En sistemas UNIX, Linux y Windows, puede utilizar mandatos PCF para acceder a los mandatos de administración del gestor de autorizaciones sobre objetos (OAM).

Los mandatos PCF y los mandatos gestor de autorizaciones sobre objetos equivalentes son los siguientes:

Tabla 6. 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.

Para obtener más información sobre cómo utilizar estos mandatos, consulte Introducción a los formatos de mandatos programables .

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 WebSphere 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) SSL o 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.

Seguridad de objetos en sistemas 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 WebSphere MQ.

Siempre debe ejecutar amqcrsta con el ID de usuario "mqm".

ID de usuario en sistemas 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.

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 WebSphere 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 de 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 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 sistemas UNIX, Linux y Windows por cualquier otro motivo.

Utilización de un servicio de autorización personalizado

IBM WebSphere MQ proporciona un servicio de autorización instalable. Puede optar por instalar un servicio alternativo.

El componente del servicio de autorización proporcionado con IBM WebSphere 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 WebSphere 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 7. 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 61, 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.
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 61, 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 clientes MQI de IBM WebSphere MQ, 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.

Asegúrese de que el campo MCAUSER tenga una restricción de 12 caracteres en plataformas Windows porque los caracteres adicionales se truncarán, lo que puede conducir a errores de autorización.

El cliente IBM WebSphere 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:

- 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 WebSphere 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 WebSphere 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.

En servidores de sistemas UNIX, Linux y Windows, el contenido del campo `MCAUserIdentifier` que se recibe del cliente cambia a minúsculas.

En servidores de IBM i, el contenido del campo `LongMCAUserIdentifier` que se recibe del cliente cambia a mayúsculas.

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 transfiere cuando se utiliza una aplicación de enlace de MQ JMS es el ID de usuario para la JVM en la que se ejecuta 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 SSL o 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 62
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 68
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 los canales con SSL" en la página 74
El soporte de SSL en IBM WebSphere 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 8 en la página 63.

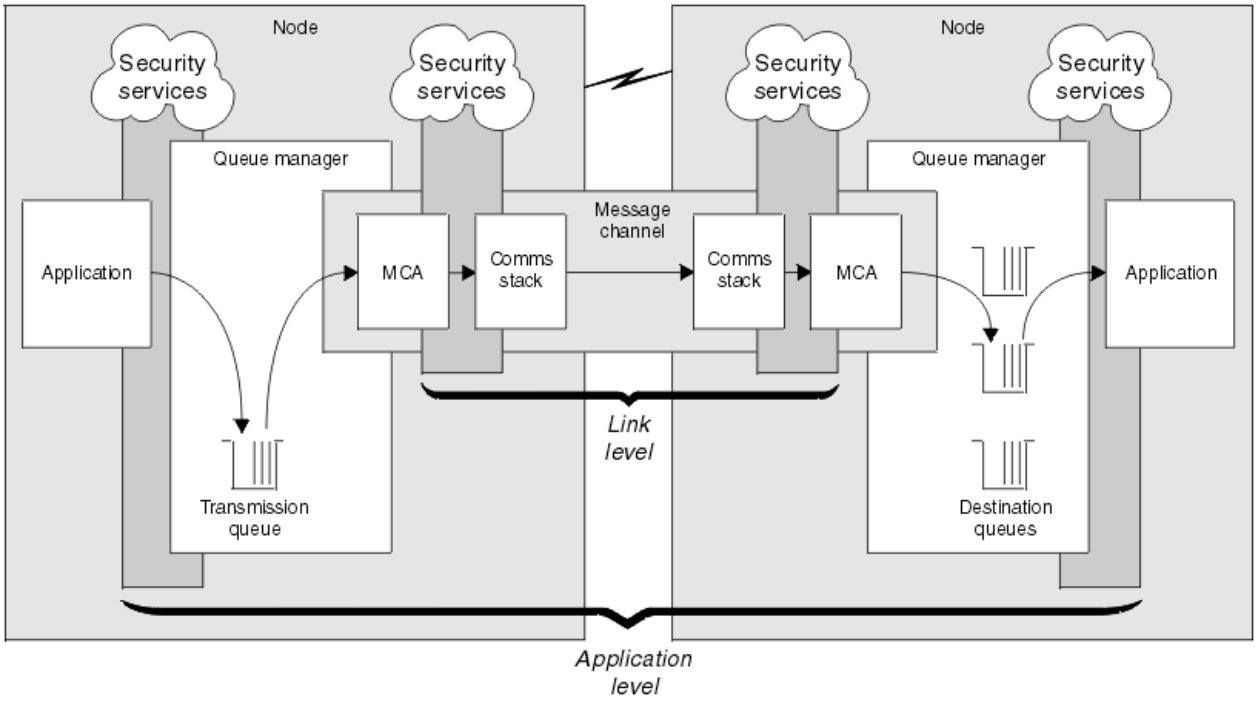


Figura 8. Seguridad a nivel de enlace y seguridad a nivel de aplicación

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.

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 8 en la página 63.

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 WebSphere 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 WebSphere 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 8 en la página 63.

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 WebSphere MQ

El principal medio de provisión de confidencialidad e integridad de datos en IBM WebSphere MQ es mediante el uso de SSL o TLS. Para obtener más información sobre el uso de SSL y TLS en IBM WebSphere MQ, consulte "Soporte de IBM WebSphere MQ para SSL y TLS" en la página 24. Para la autenticación, IBM WebSphere 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 41.

Cómo proporcionar su propia seguridad a nivel de enlace

En esta colección de temas se describe cómo puede proporcionar sus propios servicios de seguridad a 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 68. El mismo tema también describe el programa de salida de canal que se proporciona con IBM WebSphere MQ parar 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 WebSphere MQ, o una combinación de cualquiera de esos productos que funcionen conjuntamente. La seguridad a nivel de aplicación se ilustra en la Figura 8 en la página 63.

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

IBM WebSphere MQ Advanced Message Security (AMS) es un componente con licencia separada de IBM WebSphere MQ que proporciona un nivel alto de protección para los datos sensibles que fluyen por la red IBM WebSphere 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 WebSphere MQ. Sin embargo, tales soluciones requieren habilidades especiales y pueden ser complicadas y caras de mantener. IBM WebSphere MQ 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.

IBM WebSphere MQ Advanced Message Security amplía las características de seguridad de IBM WebSphere 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 WebSphere MQ.
- Se integra con IBM WebSphere MQ Managed File Transfer para proporcionar una solución de mensajería segura de principio a fin.

Para obtener más información, consulte “IBM WebSphere MQ Advanced Message Security” en la página 275.

Cómo proporcionar su propia seguridad a nivel de aplicación
En esta colección de temas se describe cómo puede proporcionar sus propios servicios de seguridad a nivel de aplicación.

Para ayudarle a implementar la seguridad a nivel de aplicación, IBM WebSphere MQ proporciona dos salidas, la salida de API y la salida cruzada de API.

Estas salidas pueden proporcionar servicios de identificación y autenticación, control de accesos, confidencialidad, integridad de datos y no rechazo, 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 WebSphere 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 WebSphere 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 9 en la página 68 y se describen en los temas siguientes.

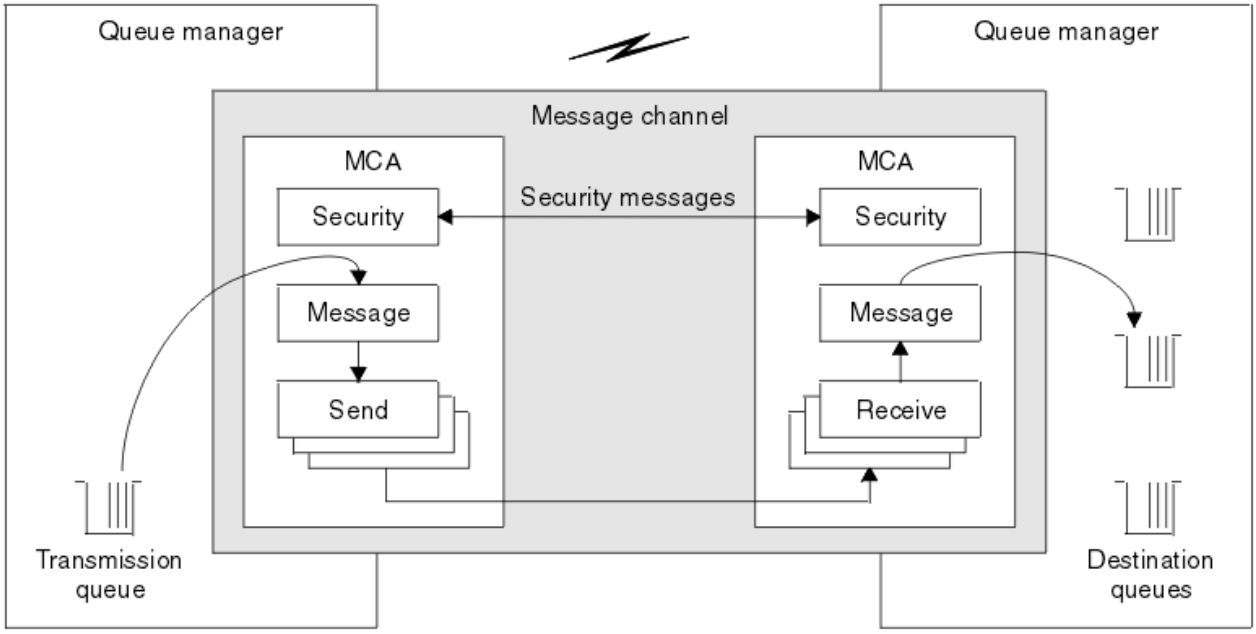


Figura 9. Salidas de seguridad, mensajes, emisión y recepción en un canal de mensajes

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 65.

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 cliente MQI de IBM WebSphere MQ 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 65.

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 WebSphere 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 WebSphere 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 65.

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, podría optar por utilizar IBM WebSphere MQ Advanced Message Security para firmar digitalmente los mensajes con el fin de protegerlos frente la modificación no autorizada. También puede utilizar los programas de salida de API si los recursos estándares no satisfacen los requisitos.

A nivel de enlace, puede optar por utilizar SSL o 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 los canales con SSL” en la página 74
El soporte de SSL en IBM WebSphere 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 datos en IBM WebSphere 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 66

IBM WebSphere MQ Advanced Message Security (AMS) es un componente con licencia separada de IBM WebSphere MQ que proporciona un nivel alto de protección para los datos sensibles que fluyen por la red IBM WebSphere MQ, aunque no afecta a las aplicaciones finales.

Referencia relacionada

Referencia a la salida de la API

Llamadas de salida de canal y estructuras de datos

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 WebSphere 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:

Cambios realizados en IBM WebSphere MQ utilizando las interfaces IBM WebSphere MQ
Configure IBM WebSphere MQ para emitir sucesos de instrumentación, específicamente sucesos de mandatos y sucesos de configuración.

Los cambios realizados en IBM WebSphere MQ fuera de su control
Algunos cambios pueden afectar a cómo se comporta IBM WebSphere MQ, pero no pueden supervisarse directamente mediante IBM WebSphere MQ. Algunos ejemplos de estos cambios incluyen cambios en los archivos de configuración `mqs.ini`, `qm.ini`y `mqclient.ini`, la creación y supresión de gestores de colas, la instalación de archivos binarios como, por ejemplo, programas de salida de usuario y cambios en los permisos de archivo. 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*.

Control operativo de IBM WebSphere 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 WebSphere MQ se puede configurar para emitir sucesos de instrumentación.

La actividad de aplicación dentro de IBM WebSphere MQ
Para auditar las acciones de aplicaciones, por ejemplo la apertura de colas y la transferencia y obtención de mensajes, configure IBM WebSphere 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 WebSphere 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 WebSphere 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, necesita un ID de usuario que tenga acceso a diversos recursos de IBM WebSphere MQ.

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 WebSphere MQ, los canales pueden estar protegidos por el soporte TLS o SSL.

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. Consulte Autorizaciones para llamadas de contexto para obtener más información.
- 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, puede abrir las colas de destino para transferir los mensajes a estas.

  Esto implica el uso de la MQI (interfaz de cola de mensajes), por lo que es posible que sea necesario realizar comprobaciones de control de acceso adicionales si no utiliza el Gestor de autorizaciones sobre objetos (OAM) de WebSphere 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 tendrá derechos administrativos completos 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.
- Si utiliza un ID de usuario que carece de privilegios administrativos de WebSphere, debe otorgar la autorización dsp y ctrlx para el canal al ese ID de usuario para que el canal funcione. 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. Por ejemplo, emita los mandatos siguientes:

```
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 41
Para ejercer un control más preciso sobre el acceso que se otorga a la conexión de los sistemas en un nivel de canal, puede utilizar los 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 WebSphere MQ no son objetos IBM WebSphere MQ; el OAM no controla el acceso a los mismos. IBM WebSphere 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 10 en la página 94.

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 68 para obtener más información sobre las salidas de canal.

Protección de los canales con SSL

El soporte de SSL en IBM WebSphere MQ utiliza el objeto de información de autenticación del gestor de colas y diversos mandatos MQSC. También debe contemplar el uso de certificados digitales.

Mandatos y atributos para soporte SSL

El protocolo SSL (Secure Sockets Layers) proporciona seguridad de canal, con protección contra escuchas y manipulaciones no autorizadas y contra falsas identidades. IBM WebSphere MQ para SSL le permite especificar, en la definición de canal, que un determinado canal utiliza seguridad SSL. 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 SSL:

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 SSL:

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/SSL mejorada.

SSLCRYP
En Windows, los sistemas UNIX and Linux establecen 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 SSL cuando un canal que utiliza SSL no pueda establecer una conexión SSL correctamente.

SSLFIPS
Especifica si sólo se deben utilizar algoritmos certificados por FIPS si el cifrado se lleva a cabo en IBM WebSphere 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 Windows, los sistemasUNIX and Linux asocian un repositorio de claves con 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) le permite utilizar la seguridad SSL en sistemas Windows, UNIX and Linux).

SSLRKEYC
El número de bytes que se deben enviar y recibir en una conversación SSL 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 SSL:

SSLCAUTH
Define si IBM WebSphere MQ requiere y valida un certificado del cliente SSL.

SSLCIPH
Especifica la potencia del cifrado y su función (CipherSpec), por ejemplo, NULL_MD5 o RC4_MD5_US. 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 iKeycmd para gestionar certificados en sistemas UNIX and Linux y la herramienta runmqakm para gestionar certificados en sistemas UNIX, Linux y 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 SSL, consulte

- “Soporte de IBM WebSphere MQ para SSL y TLS” en la página 24

Para conocer detalles de los mandatos MQSC asociados a SSL, consulte

- ALTER AUTHINFO
- DEFINE AUTHINFO
- DELETE AUTHINFO

- DISPLAY AUTHINFO

Para conocer detalles de los mandatos PCF asociados a SSL, consulte

- Cambiar, copiar y crear un objeto de información de autenticación
- Suprimir objeto de información de autenticación
- Consultar objeto de información de autenticación

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 sistemas Windows, UNIXy Linux , puede utilizar certificados autofirmados. Consulte “Creación de un certificado personal autofirmado en los sistemas UNIX, Linux, and Windows” en la página 126 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 un certificado digital adecuado.

Para obtener más información sobre la relación entre CipherSpecs y los certificados digitales, consulte “Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ” en la página 36

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 WebSphere MQ, consulte “Políticas de validación de certificados en IBM WebSphere MQ” en la página 35.

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 WebSphere 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.

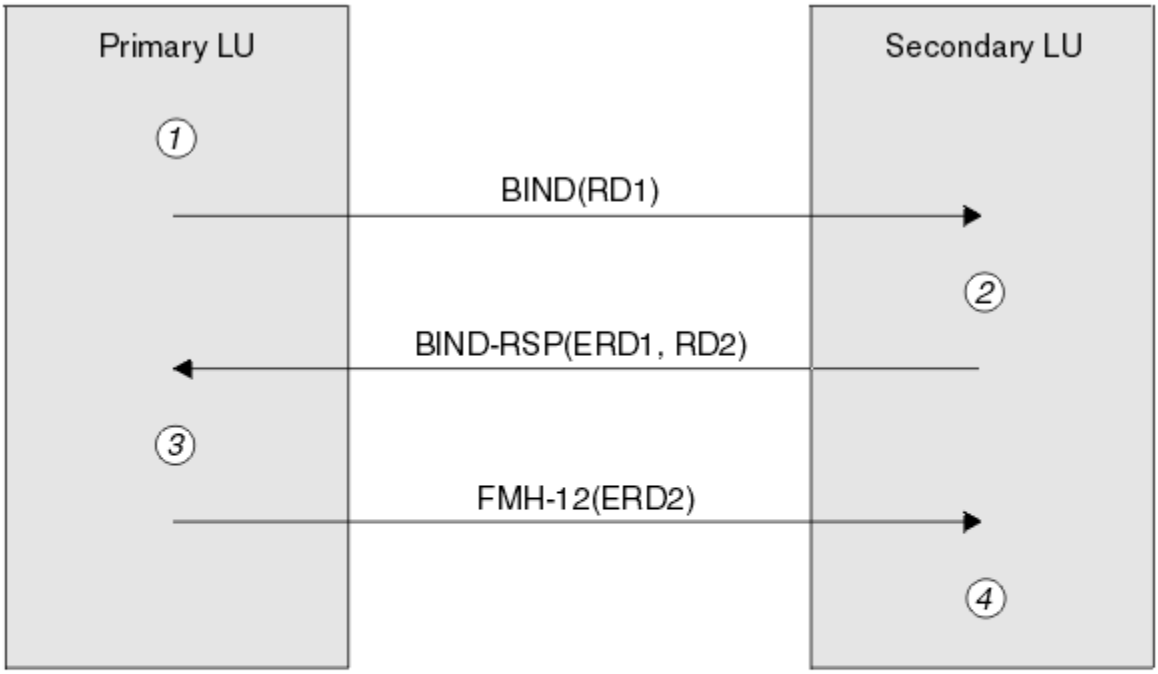
Autenticación a nivel de sesión

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 10 en la página 78 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 10. 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 10 en la página 78.

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 WebSphere 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 el manual *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.

Soporte para la autenticación del nivel de conversación en IBM WebSphere MQ en UNIX y Windows.
Lea este tema para obtener una visión general de cómo funciona la autenticación a nivel de conversación en UNIX, Linux, and Windows.

El soporte para la autenticación a nivel de conversación en IBM WebSphere MQ para WebSphere MQ en sistemas UNIX y WebSphere MQ para Windows se ilustra en la Figura 11 en la página 80. Los números del diagrama corresponden a los números de la siguiente descripción.

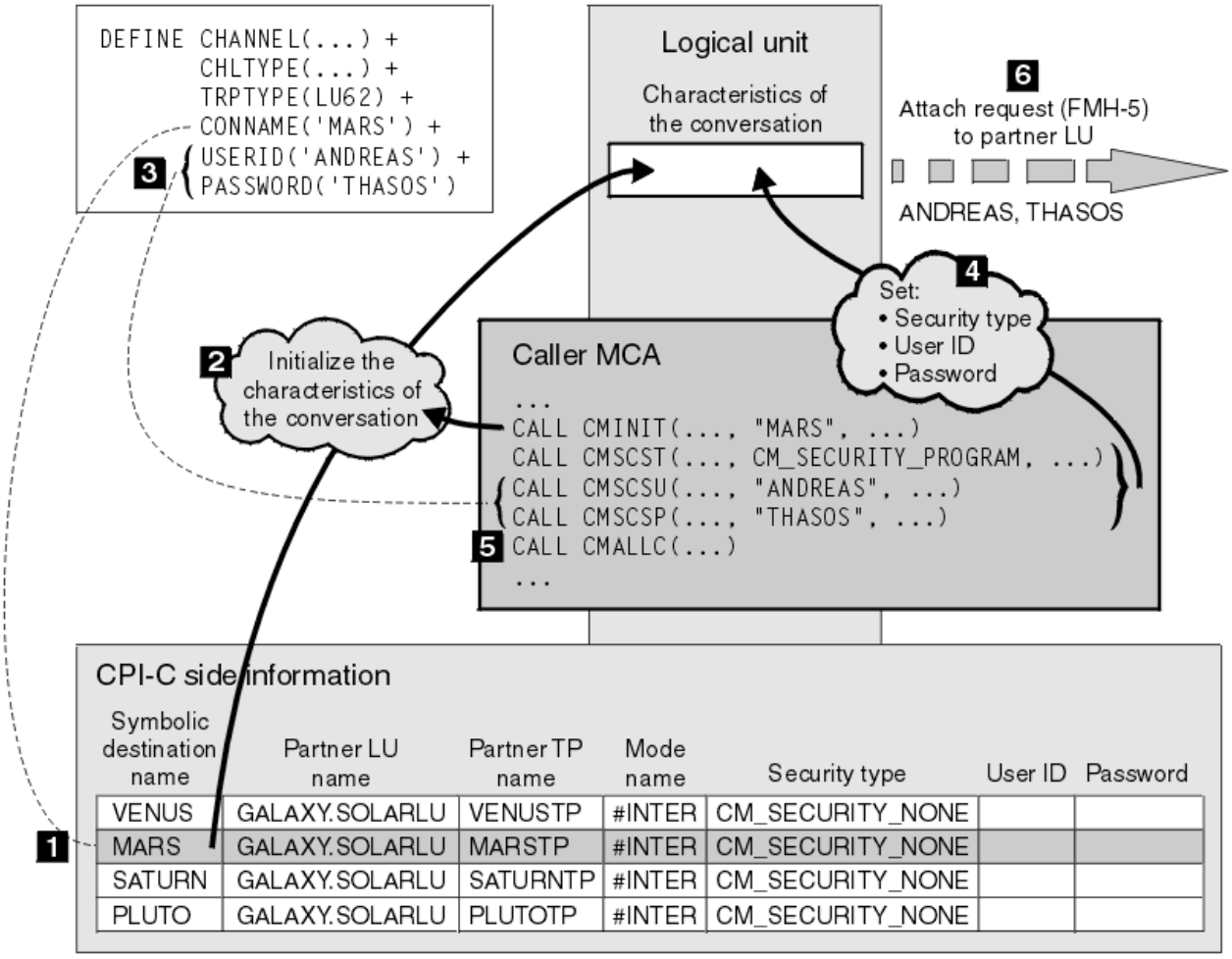


Figura 11. Soporte de WebSphere MQ para autenticación a nivel de conversación

En IBM i, sistemas UNIX y sistemas 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.

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 WebSphere MQ decide a qué gestor de colas lo envía. Cuando muchas aplicaciones envían mensajes a una cola de clúster, WebSphere 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, WebSphere 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.

Tareas relacionadas

“Cómo impedir que los gestores de colas reciban mensajes” en la página 254
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 WebSphere MQ

Existen consideraciones de seguridad adicionales si está utilizando Publicación/Suscripción de IBM WebSphere 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 WebSphere 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 WebSphere 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 WebSphere MQ Multicast.

IBM WebSphere 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 IBM WebSphere MQ (por ejemplo, las aplicaciones LLM, consulte Interoperatividad de multidifusión con mensajes de baja latencia de WebSphere 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 WebSphere MQ está basado en los ID de usuario. Para obtener más información sobre este asunto, consulte “Control de accesos para clientes” en la página 60.

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 WebSphere 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 187.

Seguridad para IBM WebSphere MQ Internet Pass-thru

Internet Pass-thru puede simplificar la comunicación a través de un cortafuegos, pero esto tiene implicaciones de seguridad.

IBM WebSphere MQ Internet Pass-thru es una ampliación del producto base IBM WebSphere MQ que se proporciona con SupportPac MS81.

WebSphere MQ Internet Pass-thru permite que dos gestores de colas intercambien mensajes, o que una aplicación cliente de WebSphere 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 WebSphere MQ y lo hace más manejable ya que canaliza los flujos en túneles mediante HTTP o actuando como servidor proxy. Mediante SSL (Capa de sockets seguros), se puede utilizar también para cifrar y descifrar los mensajes que se envían a través de Internet.

Cuando el sistema WebSphere MQ se comunique con IPT, a menos que se utilice SSLProxyMode en IPT, asegúrese de que la CipherSpec que utiliza WebSphere MQ coincida con la CipherSuite que utiliza IPT:

- Cuando IPT actúa como el servidor SSL o TLS y WebSphere MQ se conecta como el cliente SSL o TLS, la CipherSpec que utiliza WebSphere MQ debe corresponder a una CipherSuite que esté habilitada en el conjunto de claves IPT relevante.
- Cuando IPT actúa como el cliente SSL o TLS y se conecta a un servidor SSL o TLS de WebSphere MQ, la CipherSuite de IPT debe coincidir con la CipherSpec definida en el canal WebSphere MQ receptor.

Si realiza una migración desde IPT al soporte para SSL y TLS de WebSphere MQ integrado, transfiera los certificados digitales desde IPT mediante iKeyman.

Para más información, consulte WebSphere MQ Internet Pass-Thru (SupportPac MS81).

Configuración de seguridad

Esta colección de temas contiene información específica para distintos sistemas operativos y para el uso de clientes.

Configuración de la seguridad en sistemas UNIX, Linux, and Windows

Consideraciones de seguridad específicas a sistemas UNIX, Linux, and Windows.

Los gestores de colas de IBM WebSphere 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 WebSphere MQ
Puede definir el conjunto de usuarios que puede emitir mandatos para administrar IBM WebSphere MQ.

Quién puede utilizar objetos IBM WebSphere 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 WebSphere 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 WebSphere MQ; no utilice mandatos estándar del sistema operativo para otorgar o revocar autorizaciones a los recursos de la MQI.

Conexión a IBM WebSphere MQ utilizando Terminal Services

El derecho de usuario de **Create global objects** puede causar problemas si utiliza Terminal Services.

Si se está conectando a un sistema Windows utilizando Terminal Services y tiene problemas para crear o iniciar un gestor de colas, esto puede deberse al derecho de usuario, **Create global objects**, en las versiones recientes de Windows.

El derecho de usuario de **Create global objects** limita los usuarios autorizados para crear objetos en el espacio de nombres global. Para que una aplicación pueda crear un objeto global, debe estar en ejecución en el espacio de nombres global, o el usuario bajo el que se ejecuta la aplicación debe tener aplicado el derecho de usuario **Create global objects** .

Los administradores tienen el derecho de usuario de **Create global objects** aplicado de forma predeterminada, por lo que un administrador puede crear e iniciar gestores de colas cuando está conectado utilizando Terminal Services sin alterar los derechos de usuario.

Si los distintos métodos de administración de WebSphere MQ no funcionan cuando se utilizan servicios de terminal, intente establecer el derecho de usuario **Create global objects** :

1. Abra el panel Herramientas administrativas:

   Windows 2003 y Windows XP
   Acceda a este panel utilizando **Panel de control** > **Herramientas administrativas**.

   Windows Vista y Windows Server 2008
   Acceda a este panel utilizando **Panel de control** > **Sistema y mantenimiento** > **Herramientas administrativas**.
2. Efectúe una doble pulsación en **Directiva de seguridad local**.
3. Expanda `Local Policies`.
4. Pulse `User Rights Assignment`.
5. Añada el nuevo usuario o grupo a la política **Create global objects** .

Creación y gestión de grupos en Windows

Estas instrucciones le guían a través del proceso de administrar grupos en una estación de trabajo o una máquina servidor de miembros.

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.

Cualquier cambio que realice en los miembros de grupo de un principal no se reconocerá hasta que se reinicie el gestor de colas, o hasta que emita el mandato MQSC REFRESH SECURITY (o el mandato PCF equivalente).

Utilice el panel Administración de equipos para trabajar con usuarios y grupos. Puede que los cambios efectuados en la sesión iniciada actual no sean efectivos hasta que se vuelva a iniciar la sesión.

Windows 2003 y Windows XP
Acceda a este panel utilizando **Panel de control** > **Herramientas administrativas** > **Administración de equipos**.

Windows Vista y Windows Server 2008
Acceda a este panel utilizando el **Panel de control** > **Sistema y mantenimiento** > **Herramientas administrativas** > **Administración de equipos**.

Windows 7
Acceda a este panel utilizando **Herramientas administrativas** > **Administración de equipos**

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**.

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.

Visualización de quién está en 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.

Eliminación 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.

Creación y gestión de grupos en HP-UX

En HP-UX, siempre y cuando no esté utilizando NIS o NIS+, utilice la herramienta System Administration Manager (SAM) para trabajar con grupos.

Creación de un grupo en HP-UX

Añada un usuario a un grupo utilizando la herramienta System Administration Manager

Procedimiento

1. Desde SAM (System Administration Manager), efectúe una doble pulsación en Cuentas para usuarios y grupos.
2. Efectúe una doble pulsación en Grupos.
3. Seleccione Añadir en el desplegable Acciones para visualizar el panel Añadir un grupo nuevo.
4. Escriba el nombre del grupo y seleccione los usuarios que desea añadir al grupo.
5. Pulse Aplicar para crear el grupo.

Resultados

Ya ha creado un grupo.

Adición de un usuario a un grupo en HP-UX

Añada un usuario a un grupo utilizando la herramienta System Administration Manager.

Procedimiento

1. Desde SAM (System Administration Manager), efectúe una doble pulsación en Cuentas para usuarios y grupos.
2. Efectúe una doble pulsación en Grupos.
3. Resalte el nombre del grupo y seleccione Modificar en el desplegable Acciones para visualizar el panel Modificar un grupo existente.
4. Seleccione un usuario que desee añadir el grupo y pulse Añadir.
5. Si desea añadir otros usuarios al grupo, repita el paso 4 para cada usuario.
6. Cuando haya terminado de añadir nombres a la lista, pulse Aceptar.

Resultados

Ya ha añadido un usuario a un grupo.

Visualización de quién está en un grupo en HP-UX

Visualice los miembros de un grupo utilizando la herramienta System Administration Manager.

Procedimiento

1. Desde SAM (System Administration Manager), efectúe una doble pulsación en Cuentas para usuarios y grupos.
2. Efectúe una doble pulsación en Grupos.
3. Resalte el nombre del grupo y seleccione Modificar en el desplegable Acciones para visualizar el panel Modificar un grupo existente, que muestra una lista de los usuarios del grupo.

Resultados

Se muestran los miembros del grupo.

Eliminación de un usuario de un grupo en HP-UX

Elimine un usuario de un grupo utilizando la herramienta System Administration Manager.

Procedimiento

1. Desde SAM (System Administration Manager), efectúe una doble pulsación en Cuentas para usuarios y grupos.
2. Efectúe una doble pulsación en Grupos.
3. Resalte el nombre del grupo y seleccione Modificar en el desplegable Acciones para visualizar el panel Modificar un grupo existente.

4. Seleccione un usuario que desee eliminar del grupo y pulse Eliminar.
5. Si desea eliminar otros usuarios del grupo, repita el paso 4 para cada usuario.
6. Cuando haya terminado de eliminar nombres de la lista, pulse Aceptar.

Resultados
Ya ha eliminado un usuario de un grupo.

Creación y gestión de grupos en AIX

En AIX, siempre y cuando no esté utilizando NIS o NIS+, utilice SMITTY para trabajar con grupos.

Creación de un grupo
Crear un grupo utilizando SMITTY.

Procedimiento

1. En SMITTY, seleccione Seguridad y usuarios y pulse Intro.
2. Seleccione Grupos y pulse Intro.
3. Seleccione Añadir un grupo y pulse Intro.
4. Escriba el nombre del grupo y los nombres de los usuarios que desee añadir al grupo, separados por comas.
5. Pulse Intro para crear el grupo.

Resultados
Ya ha creado un grupo.

Adición de un usuario a un grupo
Añada un usuario a un grupo utilizando SMITTY.

Procedimiento

1. En SMITTY, seleccione Seguridad y usuarios y pulse Intro.
2. Seleccione Grupos y pulse Intro.
3. Seleccione Cambiar/Mostrar características de un grupo y pulse Intro.
4. Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.
5. Añada los nombres de los usuarios que desea añadir al grupo, separados por comas.
6. Pulse Intro para añadir los nombres al grupo.

Visualización de los miembros de un grupo
Visualizar quién está en un grupo utilizando SMITTY.

Procedimiento

1. En SMITTY, seleccione Seguridad y usuarios y pulse Intro.
2. Seleccione Grupos y pulse Intro.
3. Seleccione Cambiar/Mostrar características de un grupo y pulse Intro.
4. Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.

Resultados
Se muestran los miembros del grupo.

Supresión de un usuario de un grupo
Elimine un usuario de un grupo utilizando SMITTY.

Procedimiento

1. En SMITTY, seleccione Seguridad y usuarios y pulse Intro.
2. Seleccione Grupos y pulse Intro.
3. Seleccione Cambiar/Mostrar características de un grupo y pulse Intro.
4. Escriba el nombre del grupo para que aparezca una lista de los miembros del grupo.
5. Suprima los nombres de los usuarios que desea eliminar del grupo.
6. Pulse Intro para eliminar los nombres del grupo.

Resultados

Ya ha eliminado un usuario de un grupo.

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.

Creación de un grupo en Solaris

Crear un grupo mediante el mandato **groupadd**.

Procedimiento

Escriba el siguiente mandato: `groupadd` *`group-name`*

donde *nombre-grupo* es el nombre del grupo.

Resultados

El archivo `/etc/group` contiene información sobre los grupos.

Adición de un usuario a un grupo en Solaris

Añada un usuario a un grupo mediante el mandato **usermod**.

Procedimiento

Para añadir un miembro a un grupo adicional, ejecute el mandato `usermod` y liste 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 es miembro del grupo `groupa`y se va a convertir también en miembro de `groupb` , utilice el mandato siguiente: `usermod -G groupa,groupb` *`user-name`* , donde *nombre-usuario* es el nombre de usuario.

Visualización de quién está en un grupo en Solaris

Para descubrir quién es miembro de un grupo, examine la entrada de dicho grupo en el archivo `/etc/group`.

Eliminación de un usuario de un grupo en Solaris

Elimine un usuario de un grupo mediante el mandato **usermod**.

Procedimiento

Para eliminar un miembro de un grupo adicional, ejecute el mandato **usermod**, que lista 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` , se utiliza el mandato siguiente: `usermod -G groupa,groupb` *`user-name`*, donde *`user-name`* es el nombre de usuario.

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.

Creación de un grupo en Linux

Crear un grupo mediante el mandato **groupadd**.

Procedimiento

Para crear un grupo nuevo, escriba el mandato siguiente: `groupadd -g group-ID group-name`

, donde *ID-grupo* es el identificador del grupo y *nombre-grupo* es el nombre del grupo.

Resultados

El archivo `/etc/group` contiene información sobre los grupos.

Adición de un usuario a un grupo en Linux

Añada un usuario a un grupo mediante el mandato **usermod**.

Procedimiento

Para añadir un miembro a un grupo adicional, ejecute el mandato `usermod` y liste 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 es miembro del grupo `groupay` se va a convertir también en miembro de `groupb` , se utiliza el mandato siguiente: `usermod -G groupa,groupb user-name`

, donde *user-name* es el nombre de usuario.

Visualización de los miembros de un grupo en Linux

Visualizar quién está en un grupo mediante el mandato **getent**.

Procedimiento

Para visualizar quién es miembro de un grupo, escriba el mandato siguiente: `getent group group-name`

, donde *nombre-grupo* es el nombre del grupo.

Supresión de un usuario de un grupo

Elimine un usuario de un grupo mediante el mandato **usermod**.

Procedimiento

Para eliminar un miembro de un grupo adicional, ejecute el mandato **usermod**, que lista 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` , se utiliza el mandato siguiente: `usermod -G groupa,groupb user-name`

, donde *nombre-usuario* es el nombre de usuario.

Cómo funcionan las autorizaciones

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.

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 8 en la página 93, Tabla 9 en la página 93, Tabla 10 en la página 94 y Tabla 11 en la página 94 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 8. Autorización de seguridad necesaria para llamadas MQCONN			
Autorización necesaria para:	**Objeto de cola (“1” en la página 94)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCONN	No aplicable	No aplicable	MQZAO_CONNECT

Tabla 9. Autorización de seguridad necesaria para llamadas MQOPEN			
Autorización necesaria para:	**Objeto de cola (“1” en la página 94)**	**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 94)**	MQZAO_INPUT	No aplicable	No aplicable
MQOO_OUTPUT (Cola normal) **(“3” en la página 94)**	MQZAO_OUTPUT	No aplicable	No aplicable
MQOO_PASS_ IDENTITY_CONTEXT **(“4” en la página 94)**	MQZAO_PASS_ IDENTITY_CONTEXT	No aplicable	No se comprueba
MQOO_PASS_ALL_ CONTEXT **(“4” en la página 94, “5” en la página 94)**	MQZAO_PASS _ALL_CONTEXT	No aplicable	No se comprueba
MQOO_SET_ IDENTITY_CONTEXT **(“4” en la página 94, “5” en la página 94)**	MQZAO_SET_ IDENTITY_CONTEXT	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“6” en la página 94)**
MQOO_SET_ ALL_CONTEXT **(“4” en la página 94, “7” en la página 94)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 94)**
MQOO_OUTPUT (cola de transmisión) **(“8” en la página 94)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 94)**
MQOO_SET	MQZAO_SET	No aplicable	No se comprueba
MQOO_ALTERNATE_ USER_AUTHORITY	**(“9” en la página 95)**	**(“9” en la página 95)**	MQZAO_ALTERNATE_ USER_AUTHORITY **(“9” en la página 95, “10” en la página 95)**

Tabla 10. Autorización de seguridad necesaria para llamadas MQPUT1			
Autorización necesaria para:	**Objeto de cola (“1” en la página 94)**	**Objeto de proceso**	**Objeto gestor de colas**
MQPMO_PASS_ IDENTITY_CONTEXT	MQZAO_PASS_ IDENTITY_CONTEXT **(“11” en la página 95)**	No aplicable	No se comprueba
MQPMO_PASS_ALL _CONTEXT	MQZAO_PASS_ ALL_CONTEXT **(“11” en la página 95)**	No aplicable	No se comprueba
MQPMO_SET_ IDENTITY_CONTEXT	MQZAO_SET_ IDENTITY_CONTEXT **(“11” en la página 95)**	No aplicable	MQZAO_SET_ IDENTITY_CONTEXT **(“6” en la página 94)**
MQPMO_SET_ ALL_CONTEXT	MQZAO_SET_ ALL_CONTEXT **(“11” en la página 95)**	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 94)**
(Cola de transmisión) **(“8” en la página 94)**	MQZAO_SET_ ALL_CONTEXT	No aplicable	MQZAO_SET_ ALL_CONTEXT **(“6” en la página 94)**
MQPMO_ALTERNATE_ USER_AUTHORITY	**(“12” en la página 95)**	No aplicable	MQZAO_ALTERNATE_ USER_AUTHORITY **(“10” en la página 95)**

Tabla 11. Autorización de seguridad necesaria para llamadas MQCLOSE			
Autorización necesaria para:	**Objeto de cola (“1” en la página 94)**	**Objeto de proceso**	**Objeto gestor de colas**
MQCO_DELETE	MQZAO_DELETE **(“13” en la página 95)**	No aplicable	No aplicable
MQCO_DELETE _PURGE	MQZAO_DELETE **(“13” en la página 95)**	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 94).
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 sólo se lleva a cabo si se cumplen las dos condiciones 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 que se utiliza.

    De lo contrario, no hay comprobación.

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 100)

Objeto	Autorización necesaria
Cola	MQZAO_CREATE **(“2” en la página 100)**
Tema	MQZAO_CREATE **(“2” en la página 100)**
Proceso	MQZAO_CREATE **(“2” en la página 100)**
Gestor de colas	No aplicable
Lista de nombres	MQZAO_CREATE **(“2” en la página 100)**
Información de autenticación	MQZAO_CREATE **(“2” en la página 100)**
Canal	MQZAO_CREATE **(“2” en la página 100)**
Canal de conexión de cliente	MQZAO_CREATE **(“2” en la página 100)**
Escucha	MQZAO_CREATE **(“2” en la página 100)**
Servicio	MQZAO_CREATE **(“2” en la página 100)**
Información de comunicación	MQZAO_CREATE **(“2” en la página 100)**

DEFINE *objeto* REPLACE (“1” en la página 100, “3” en la página 100)

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
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
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

Objeto	Autorización necesaria
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
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
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

Mandato	Objeto	Autorización necesaria
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 en DEFINE *objeto* NOREPLACE.

Información relacionada

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

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

Borrar *objeto*

Objeto	Autorización necesaria
Cola	MQZAO_CLEAR

Objeto	Autorización necesaria
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

Copiar *objeto* (sin sustituir) (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)**
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 106)**

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

Objeto	Autorización necesaria
Servicio	MQZAO_CHANGE
Información de comunicación	MQZAO_CHANGE

Crear *objeto* (sin sustituir) (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
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

Objeto	Autorización necesaria
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
Información de comunicación	MQZAO_DISPLAY

Inquire nombres *objeto*

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

Detenga *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

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

Mandato	Objeto	Autorización necesaria
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

Mandato	Objeto	Autorización necesaria
Renovar clúster	grupo de pertenencia 'mqm' necesario	
Restablecer clúster	grupo de pertenencia 'mqm' necesario	
Suspender clúster de gestores de colas	grupo de pertenencia 'mqm' necesario	
Reanudar clúster de gestores de colas	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.

Consideraciones especiales para la seguridad en Windows

Algunas funciones de seguridad se comportan de forma diferente en las distintas versiones de Windows.

La seguridad de IBM WebSphere 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. Este conjunto de temas incluye descripciones de cómo estas diferencias pueden afectar a la seguridad de IBM WebSphere MQ cuando se ejecuta IBM WebSphere MQ en un entorno Windows.

El programa de salida de canal SSPI

WebSphere MQ para 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
Este utiliza el soporte para 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 WebSphere MQ Versión 7.0.

Este servicio se suele utilizar en un canal MQI para permitir que un servidor de gestor de colas autentique un cliente MQI de WebSphere MQ. 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 WebSphere MQ Versión 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 cliente MQI de WebSphere MQ 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.

Cuando se recibe un error de 'grupo no encontrado' en Windows

Este problema puede surgir porque WebSphere MQ pierde el acceso al grupo mqm local cuando los servidores Windows se promocionan, o se degradan, a controladores de dominio. Para solucionar este problema, vuelva a crear el 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.
```

Alterar el estado de una máquina entre el servidor y el controlador de dominio afecta al funcionamiento de WebSphere MQ, ya que WebSphere MQ utiliza un grupo mqm definido localmente. 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.

Para corregir este problema, vuelva a crear el grupo mqm utilizando las herramientas de gestión de Windows estándar. Puesto que toda la información de pertenencia a grupos se pierde, debe volver a incluir los usuarios de WebSphere 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 WebSphere MQ de dominio con privilegios el nivel de autorización necesario.

Cuando surgen problemas con IBM WebSphere MQ y los controladores de dominio en Windows

Pueden surgir ciertos problemas con los valores de seguridad cuando servidores Windows se promocionan a controladores de dominio.

Al promocionar servidores Windows 2000, Windows 2003 o Windows Server 2008 para que pasen a ser controladores de dominio, se le ofrecerá la opción de seleccionar un valor de seguridad predeterminado o no predeterminado relacionado con los permisos de usuario y de grupo. Esta opción controla si los usuarios arbitrarios pueden recuperar miembros de grupo desde Active Directory. Puesto que WebSphere MQ depende de la información de pertenencia a grupos para implementar su política de seguridad, es importante que el ID de usuario que está realizando las operaciones WebSphere MQ pueda determinar la pertenencia a grupos de otros usuarios.

En Windows 2000, cuando se crea un dominio utilizando la opción de seguridad predeterminada, el ID de usuario predeterminado creado por WebSphere MQ durante el proceso de instalación puede obtener las pertenencias a grupos de otros usuarios cuando sea necesario. El producto se instala luego normalmente, creando los objetos predeterminados, y el gestor de colas puede determinar la autorización de acceso de los usuarios locales y de dominio, si es necesario.

En Windows 2000, cuando se crea un dominio utilizando la opción de seguridad no predeterminada, o en Windows 2003 y Windows Server 2008 cuando se crea un dominio utilizando la opción de seguridad predeterminada, el ID de usuario creado por WebSphere MQ durante la instalación no siempre puede determinar las pertenencias a grupos necesarias. En este caso, debe saber:

- Cómo se comporta Windows 2000 con permisos de seguridad no predeterminados, o cómo se comportan Windows 2003 y Windows Server 2008 con permisos de seguridad predeterminados
- Cómo permitir que los miembros del grupo mqm de dominio lean información de pertenencia a grupos
- Cómo configurar un servicio de IBM WebSphere MQ Windows para que se ejecute bajo un usuario de dominio

Dominio de Windows 2000 con permisos de seguridad no predeterminados, o dominio de Windows 2003 y Windows Server 2008 con permisos de seguridad predeterminados

La instalación de WebSphere MQ se comporta de un modo distinto en estos sistemas operativos dependiendo de si es un usuario local o un usuario de dominio el que realiza la instalación.

Si un usuario **local** instala WebSphere MQ, el asistente de preparación de WebSphere MQ detecta que el usuario local creado para el servicio de IBM WebSphere MQ Windows puede recuperar la información de pertenencia a grupos del usuario que realiza la instalación. El Asistente de preparación de WebSphere MQ solicita al usuario información sobre la configuración de red para determinar si hay o no otras cuentas de usuario definidas en controladores de dominio que se ejecutan en Windows 2000 o posterior. De esta forma, el servicio de IBM WebSphere MQ Windows se debe ejecutar bajo una cuenta de usuario de dominio con autoridades y valores particulares. El Asistente de preparación de WebSphere MQ solicita al usuario detalles de la cuenta de este usuario. La ayuda en línea del asistente proporciona información detallada de la cuenta de usuario de dominio necesaria que se puede enviar al administrador del dominio.

Si un usuario de **dominio** instala WebSphere MQ, el asistente de preparación de WebSphere MQ detecta que el usuario local creado para el servicio de IBM WebSphere MQ Windows no puede recuperar la información de pertenencia a grupos del usuario que realiza la instalación. En este caso, el asistente de preparación de WebSphere MQ siempre solicita al usuario los detalles de cuenta de la cuenta de usuario de dominio para que los utilice el servicio de IBM WebSphere MQ Windows.

Cuando el servicio de IBM WebSphere MQ Windows necesita utilizar una cuenta de usuario de dominio, WebSphere MQ no puede funcionar correctamente hasta que se haya configurado utilizando el asistente

de preparación de WebSphere MQ. El Asistente de preparación de WebSphere MQ no permite al usuario continuar con otras tareas, hasta que el servicio de Windows se haya configurado mediante una cuenta adecuada.

Si un dominio de Windows 2000 se ha configurado con permisos de seguridad que no son los predeterminados, la solución habitual para permitir que WebSphere MQ funcione correctamente es configurarlo con una cuenta de usuario de dominio adecuada, como se ha descrito más arriba.

Consulte Creación y configuración de cuentas de dominio para WebSphere MQ para obtener más información.

Configuración IBM WebSphere MQ Servicios para ejecutar en un usuario de dominio en Windows
Utilice el Asisten de preparación de IBM WebSphere MQ para entrar los detalles de cuenta de la cuenta de usuario de dominio. De forma alternativa, puede utilizar el panel Administración de equipos para modificar los detalles de **Inicio de sesión** para el servicio de IBM WebSphere MQ específico de la instalación.

Para obtener más información, consulte Cambio de la contraseña de la cuenta de usuario de servicio de IBM WebSphere MQ Windows

Aplicación de archivos de plantilla de seguridad a Windows

La aplicación de una plantilla podría afecta a los valores de seguridad aplicados a archivos y directorios de WebSphere MQ. Si utiliza la plantilla de alta seguridad, aplíquela antes de instalar WebSphere MQ.

Windows soporta archivos de plantilla de seguridad basados en texto que se pueden 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 podría afectar a los valores de seguridad aplicados a los archivos y directorios de WebSphere MQ. Si desea utilizar la plantilla de alta seguridad, configure la máquina antes de instalar WebSphere MQ.

Si aplica la plantilla de alta seguridad a una máquina en la que ya está instalada WebSphere MQ , se eliminarán todos los permisos que haya establecido en los archivos y directorios de WebSphere MQ. 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.

Grupos anidados

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 WebSphere 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 Windows 2003 están en el nivel funcional *Windows 2000 mixto*. (Windows Server 2003 , Windows XP, Windows Vista y Windows Server 2008 siguen todos 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 WebSphere MQ. Las API de red que utiliza WebSphere MQ no dan soporte a todas las configuraciones a las que da soporte el nivel funcional del dominio. Como resultado, WebSphere 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.

Configuración de autorización adicional para aplicaciones Windows que se conectan a IBM WebSphere MQ

Es posible que la cuenta bajo la que se ejecutan los procesos IBM WebSphere MQ necesite autorización adicional antes de poder otorgar el acceso SYNCHRONIZE a procesos de aplicaciones.

Es posible que experimente problemas si tiene aplicaciones Windows, por ejemplo páginas ASP, que se conectan a IBM WebSphere MQ y que están configuradas para ejecutarse a un nivel de seguridad superior al habitual.

IBM WebSphere MQ requiere acceso SYNCHRONIZE para los procesos de aplicaciones a fin de coordinar ciertas acciones. El APAR IC35116 modificó IBM WebSphere MQ para que se especifiquen los privilegios adecuados. No obstante, es posible que la cuenta bajo la que se ejecutan los procesos IBM WebSphere MQ necesite autorización adicional antes de poder otorgar el acceso solicitado.

Cuando una aplicación de servidor intenta por primera vez conectarse a un gestor de colas de IBM WebSphere MQ modificará el acceso para otorgar autorización SYNCHRONIZE para administradores de IBM WebSphere MQ. Para configurar autorización adicional para el ID de usuario bajo el que se ejecutan los procesos de IBM WebSphere MQ, realice los pasos siguientes:

1. Inicie la herramienta Política de seguridad local, haga clic en Configuración de seguridad-> Políticas locales-> Asignaciones de derechos de usuario, haga clic en "Depurar programas".
2. Efectúe una doble pulsación en "Depurar programas" y luego añada su ID de usuario de IBM WebSphere 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.

Configuración de la seguridad en HP Integrity NonStop Server

Consideraciones de seguridad específicas a sistemas HP Integrity NonStop Server.

El cliente de IBM WebSphere MQ para HP Integrity NonStop Server soporta ambos protocolos, TLS (seguridad de la capa de transporte) y SSL (capa de sockets seguros), para proporcionar una seguridad de nivel de enlace cuando se conecte a un gestor de colas. Estos protocolos están soportados mediante el uso de una implementación de OpenSSL. OpenSSL requiere un origen de datos aleatorios para proporcionar operaciones de criptografía robusta.

OpenSSL

Visión general de la seguridad de OpenSSL para el cliente de IBM WebSphere MQ para HP Integrity NonStop Server.

El kit de herramientas de OpenSSL es una implementación de código abierto de los protocolos de Capa de sockets seguros (SSL) y del protocolo Transport Layer Security (TLS) que aseguran las comunicaciones en la red.

El kit de herramientas ha sido desarrollado por OpenSSL Project. Si desea más información sobre OpenSSL Project, consulte https://www.openssl.org. El cliente IBM WebSphere MQ para HP Integrity NonStop Server contiene versiones modificadas de las bibliotecas de OpenSSL y el mandato **openssl**. Las bibliotecas y el mandato **openssl** se trasladan del kit de herramientas OpenSSL 1.0.1c, y sólo se proporcionan como código de objeto. No se ha proporcionado código abierto.

Los programas de la aplicación cliente IBM WebSphere MQ cargan las bibliotecas de OpenSSL dinámicamente según sea necesario. Sólo las bibliotecas de OpenSSL proporcionadas por IBM WebSphere MQ están soportadas para ser utilizadas con las aplicaciones cliente IBM WebSphere MQ.

El mandato **openssl** , que se puede utilizar con fines de gestión de certificados, se instala en el directorio OSS `opt_installation_path/opt/mqm/bin`.

Utilizando el mandato **openssl**, puede crear y gestionar claves y certificados digitales con distintos formatos de datos comunes y llevar a cabo tareas sencillas de autoridad de certificación (CA).

El formato predeterminado para los datos de clave y certificado que procesa OpenSSL es el formato PEM (correo con privacidad mejorada). Los datos en el formato PEM son datos ASCII con codificación base64. Por lo tanto, los datos se pueden transferir utilizando sistemas basados en texto como, por ejemplo, correo electrónico, y se pueden cortar y pegar utilizando editores de texto y navegadores web. PEM es un estándar de Internet para los intercambios criptográficos basados en texto y se especifica en los RFC de Internet 1421, 1422, 1423 y 1424. IBM WebSphere MQ presupone que un archivo con extensión `.pem` contiene datos en formato PEM. Los archivos con formato PEM pueden contener varios certificados y otros objetos cifrados, y pueden incluir comentarios.

El soporte SSL de IBM WebSphere MQ en otros sistemas operativos podría requerir que los datos de claves y certificados se codifiquen mediante las DER (Reglas de codificación distinguida). DER es un conjunto de reglas de codificación para utilizar la notación ASN.1 en comunicaciones seguras. Los datos codificados mediante el uso de DER son datos binarios y el formato de los datos de clave y certificado codificados mediante el uso de DER también se conocen como PKCS#12 o PFX. Un archivo que contiene estos datos normalmente tiene una extensión de `.p12` o `.pfx`. El mandato **openssl** se puede convertir entre el formato PEM y PKCS#12.

Daemon de entropía

OpenSSL requiere un origen de datos aleatorios para proporcionar operaciones de criptografía robusta. La generación de números aleatorios es una prestación que, normalmente, proporciona el sistema operativo o un proceso de daemon de nivel de sistema. El sistema operativo HP Integrity NonStop Server no proporciona esta prestación en el sistema operativo.

Si utiliza el soporte SSL y TLS proporcionado con el cliente de IBM WebSphere MQ para HP Integrity NonStop Server, es necesario un proceso que se llama daemon de entropía para proporcionar el origen de datos aleatorios. Cuando se inicia un canal de cliente que requiere SSL o TLS, OpenSSL espera que un daemon de entropía se esté ejecutando y proporcionando sus servicios en un socket en el sistema de archivos OSS en `/etc/egd-pool`.

El cliente de IBM WebSphere MQ para HP Integrity NonStop Server no proporciona un daemon de entropía. El cliente de IBM WebSphere MQ para HP Integrity NonStop Server se prueba con los daemons de entropía siguientes:

- amqjkdm0 (tal como lo ha proporcionado el servidor IBM WebSphere MQ 5.3)
- `/usr/local/bin/prngd` (Versión 0.9.27, tal como lo ha proporcionado la biblioteca técnica de código abierto de HP Integrity NonStop Server)

Configuración de la seguridad del cliente MQI de IBM WebSphere MQ

Debe tener en cuenta la seguridad del cliente MQI de IBM WebSphere MQ 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:

  Debe utilizar la autenticación mutua dentro de SSL o TLS. Para obtener más información, consulte “Trabajar con SSL o TLS” en la página 114
- 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 164.

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 confirmado por el cliente con un ID de usuario MCAUSER" en la página 190.

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 41.

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 SSL o 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 SSL o TLS activa y se especifica un valor FipsRequired válido en una llamada MQCONNX de SSL, todas las conexiones SSL 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 SSL o 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 WebSphere 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. WebSphere MQ no garantiza rechazar todas las conexiones de este tipo y es responsabilidad del usuario determinar si la configuración es compatible con WebSphere MQ.

Conceptos relacionados

"Federal Information Processing Standards (FIPS) para UNIX, Linuxy Windows" en la página 27

Cuando la criptografía es necesaria en un canal SSL o TLS en Windows, sistemas UNIX and Linux , WebSphere 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 FIPS (Federal Information Processing Standards) Cryptomodule Validation Program del US National Institute of Standards and Technology, en el nivel 140-2.

Stanza SSL del archivo de configuración de cliente

Referencia relacionada

FipsRequired (MQLONG)

MQSSLFIPS

Ejecución de aplicaciones cliente SSL o TLS con varias instalaciones de GSKit V8.0 en AIX

Las aplicaciones cliente SSL o 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 SSL o 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 causado que el cliente IBM WebSphere MQ cargue un conjunto combinado de bibliotecas de dos instalaciones GSKit V8.0 diferentes. La ejecución de una aplicación cliente de IBM WebSphere MQ en un entorno de Db2 puede producir este error.

Para evitarlo, incluya los directorios de la biblioteca de IBM WebSphere MQ en la parte frontal de la vía de acceso de bibliotecas para que las bibliotecas de IBM WebSphere 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 WebSphere MQ)

Configuración de comunicaciones para SSL o TLS en sistemas UNIX, Linux, and 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.

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 o TLS en sistemas UNIX, Linux, and Windows” en la página 118.

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 WebSphere MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En los sistemas UNIX, Linux y Windows, el cliente SSL o TLS solo envía un certificado si tiene uno etiquetado con el formato correcto de IBM WebSphere MQ:

- 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 WebSphere MQ , `ibmwebspheremq` seguido por el ID de usuario de inicio de sesión cambiado a minúsculas, por ejemplo, `ibmwebspheremqmyuserid`.

IBM WebSphere 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 más información, consulte “Conexión de dos gestores de colas utilizando SSL o TLS” en la página 212.

Trabajar con SSL o TLS

Estos temas proporcionan instrucciones para realizar tareas individuales relacionados con la utilización de SSL o TLS con IBM WebSphere 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 149
- “Autorización del acceso a objetos” en la página 164
- “Confidencialidad de mensajes” en la página 211
- “Integridad de datos de mensajes” en la página 232
- “Mantenimiento de la seguridad de los clústeres” en la página 250

Trabajar con SSL o TLS en HP Integrity NonStop Server

Describe la implementación de seguridad OpenSSL del cliente IBM WebSphere MQ para HP Integrity NonStop Server, incluidos servicios de seguridad, componentes, versiones de protocolo soportadas, CipherSpecs soportadas y funciones de seguridad no soportadas.

El soporte SSL y TLS de IBM WebSphere MQ proporciona los siguientes servicios de seguridad para los canales de cliente:

- Autenticación del servidor y, de forma opcional, autenticación del cliente.
- Cifrado y descifrado de los datos que fluyen por un canal.
- Comprobaciones de integridad de los datos que fluyen por un canal.

El soporte SSL y TLS proporcionado con el cliente IBM WebSphere MQ para HP Integrity NonStop Server engloba los componentes siguientes:

- Bibliotecas OpenSSL y el mandato **openssl**.
- Mandato de ocultación de contraseña de IBM WebSphere MQ, **amqrsslc**.

Los componentes necesarios siguientes para la operación del canal de cliente SSL o TLS no se proporcionan con el cliente IBM WebSphere MQ para HP Integrity NonStop Server:

- Un daemon de entropía para proporcionar un origen de datos aleatorios para la criptografía OpenSSL.

Versiones de protocolo soportadas

El cliente IBM WebSphere MQ para HP Integrity NonStop Server soporta las versiones de protocolo siguientes:

- SSL 3.0
- TLS 1.0
- TLS 1.2

CipherSpecs soportadas

El cliente IBM WebSphere MQ para HP Integrity NonStop Server soporta las versiones de CipherSpecs siguientes:

- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA
- RC4_SHA_US
- RC4_MD5_US
- TRIPLE_DES_SHA_US
- TLS_RSA_WITH_3DES_EDE_CBC_SHA (en desuso)
- DES_SHA_EXPORT1024
- RC4_56_SHA_EXPORT1024
- RC4_MD5_EXPORT
- RC2_MD5_EXPORT
- DES_SHA_EXPORT

- TLS_RSA_WITH_DES_CBC_SHA
- NULL_SHA
- NULL_MD5
- FIPS_WITH_DES_CBC_SHA
- FIPS_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_RSA_WITH_NULL_SHA256
- TLS_RSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_256_GCM_SHA384
- ECDHE_ECDSA_AES_128_CBC_SHA256
- ECDHE_ECDSA_AES_256_CBC_SHA384
- ECDHE_RSA_AES_128_CBC_SHA256
- ECDHE_RSA_AES_256_CBC_SHA384
- ECDHE_ECDSA_AES_128_GCM_SHA256
- ECDHE_ECDSA_AES_256_GCM_SHA384
- ECDHE_RSA_AES_128_GCM_SHA256
- ECDHE_RSA_AES_256_GCM_SHA384

Funciones de seguridad no soportadas

Actualmente, el cliente IBM WebSphere MQ para HP Integrity NonStop Server no soporta:

- Soporte de hardware de cifrado PKCS#11
- Comprobación de la lista de revocación de certificados LDAP
- Comprobación del protocolo de estado de certificado en línea OCSP
- FIPS 140-2, controles de suite de cifrado NSA SUITE B

Gestión de certificados

Utilice un conjunto de archivos para almacenar información de certificados digitales y de revocación de certificados.

El soporte SSL y TLS de IBM WebSphere MQ utiliza un conjunto de archivos para almacenar información de certificados digitales y de revocación de certificados. Estos archivos se encuentran en directorio especificado mediante programa a través del campo KeyRepository en la estructura pasada en la llamada a MQCONNX, mediante la variable de entorno *MQSSLKEYR* o, en la stanza SSL de `mqclient.ini` utilizando el atributo SSLKeyRepository.

La estructura MQSCO tiene prioridad sobre la variable de entorno MQSSLKEYR que tiene prioridad sobre el valor de la stanza del archivo ini.

Importante: La ubicación del repositorio de claves especifica una ubicación de directorio y no un nombre de archivo en la plataforma HP Integrity NonStop Server.

El cliente de IBM WebSphere MQ para HP Integrity NonStop Server utiliza los siguientes nombres de archivo sensibles a mayúsculas y minúsculas en la ubicación del repositorio de claves:

- “Almacén de certificados personales” en la página 117
- “Almacén de confianza de certificados” en la página 117
- “Archivo de ocultación de frase de contraseña” en la página 117
- “Archivo de lista de revocación de certificados” en la página 117

Almacén de certificados personales
El archivo del almacén de certificados personales, `cert.pem`.

Este archivo contiene el certificado personal y la clave privada cifrada para que utilice el cliente, en formato PEM. La existencia de este archivo es opcional cuando se está utilizando los canales SSL o TLS que no requieren la autenticación de cliente. Cuando la autenticación de cliente es necesaria para el canal, y se especifica SSLCAUTH(REQUIRED) en la definición del canal, este archivo debe existir y contener tanto el certificado, como la clave privada cifrada.

Los permisos de archivo se deben definir en este archivo para permitir un acceso de lectura al propietario del almacén de certificados.

Un archivo `cert.pem` con un formato correcto debe contener exactamente dos secciones con las cabeceras y los pies de página siguientes:

```
-----BEGIN PRIVATE KEY-----
Base 64 ASCII encoded private key data here
-----END PRIVATE KEY-----
```

```
-----BEGIN CERTIFICATE-----
Base 64 ASCII encoded certificate data here
-----END CERTIFICATE-----
```

La frase de contraseña para la clave privada cifrada se almacena en el archivo de ocultación de la frase de contraseña, `Stash.sth`.

Almacén de confianza de certificados
E archivo de almacén de confianza de certificados, `trust.pem`.

Este archivo contiene los certificados necesarios para validar los certificados personales utilizados por los gestores de colas a los que se conecta el cliente, en formato PEM. El almacén de confianza de certificados es obligatorio para todos los canales de cliente SSL o TLS.

Los permisos de archivo se deben definir para limitar el acceso de escritura a este archivo.

Un archivo `trust.pem` con un formato correcto debe contener una o más secciones con las cabeceras y los pies de página siguientes:

```
-----BEGIN CERTIFICATE-----
Base 64 ASCII encoded certificate data here
-----END CERTIFICATE-----
```

Archivo de ocultación de frase de contraseña
EL archivo de ocultación de la frase de contraseña, `Stash.sth`.

Este archivo es un formato binario privado para IBM WebSphere MQ y contiene la frase de contraseña cifrada que se utiliza al acceder a la clave privada que se conserva en el archivo `cert.pem`. La propia clave privada se almacena en el almacén de certificados `cert.pem`.

Para crear o modificar este archivo se debe utilizar la herramienta de línea de mandatos de IBM WebSphere MQ **amqrsslc** con el parámetro **-s**. Por ejemplo, donde el directorio `/home/alice` contiene un archivo `cert.pem`:

```
amqrsslc -s /home/alice/cert

  Enter password for Keystore /home/alice/cert.pem :
    password

  Stashed the password in file /home/alice/Stash.sth
```

Los permisos de archivo se deben definir en este archivo para permitir el acceso de lectura al propietario del almacén de certificados personales asociados.

Archivo de lista de revocación de certificados
El archivo de la lista de revocación de certificados, `crl.pem`.

Este archivo contiene las listas de revocación de certificados (CRL) que utiliza el cliente para validar los certificados digitales, en formato PEM. La existencia de este archivo es opcional. Si este archivo no está presente, no se realiza ninguna comprobación de revocaciones de certificados al validar certificados.

Los permisos de archivo se deben definir para limitar el acceso de escritura a este archivo.

Un archivo `crl.pem` con un formato correcto debe contener una o más secciones con las cabeceras y los pies de página siguientes:

```
-----BEGIN X509 CRL-----
Base 64 ASCII encoded CRL data here
-----END X509 CRL-----
```

Trabajar con SSL o TLS en sistemas UNIX, Linux, and Windows

En sistemas UNIX, Linux y Windows , el soporte SSL (Secure Sockets Layer) se instala con IBM WebSphere MQ.

Para obtener más información sobre las políticas de validación de certificados, consulte Validación de certificados y diseño de políticas de confianza.

Utilización de iKeyman, iKeycmd, runmqakm y runmqckm

En sistemas UNIX, Linux y Windows , gestione claves y certificados digitales con la GUI de iKeyman o desde la línea de mandatos utilizando iKeycmd o runmqakm.

- Para sistemas **UNIX and Linux** :
  - 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.
  - Utilice el mandato **runmqakm** 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 SSL 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 Windows y Linux x86 de 32 bits son las únicas excepciones, ya que los programas iKeyman e iKeycmd son de 32 bits en esas plataformas.

  En las plataformas siguientes, donde el JRE era de 32 bits en versiones anteriores del producto, pero sólo es de 64 bits en IBM WebSphere MQ Version 7.5, es posible que tenga que instalar controladores PKCS#11 adicionales adecuados para la modalidad de direccionamiento de **iKeyman** y **iKeycmd** JRE. Esto es debido a que el controlador PKCS#11 debe utilizar la misma modalidad de direccionamiento que el JRE. La tabla siguiente muestra las modalidades de direccionamiento de JRE de IBM WebSphere MQ de la Version 7.5.

Tabla 12. Modalidades de direccionamiento de JRE de IBM WebSphere MQ Version 7.5

Plataforma	**Modalidad de direccionamiento de JRE**
Windows (32 bits o 64 bits)	32
Linux para System x de 32 bits	32
Linux para System x de 64 bits	64
Linux para System p	64

Tabla 12. Modalidades de direccionamiento de JRE de IBM WebSphere MQ Version 7.5 (continuación)	
Plataforma	**Modalidad de direccionamiento de JRE**
Linux para System z	64
HP-UX	64
Solaris SPARC	64
Solaris x86-64	64
AIX	64

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**. Esto 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 SSL de modo que sean compatibles con FIPS, utilice el mandato **runmqakm** en lugar de los mandatos **runmqckm** o **strmqikm**.

Para solicitar el rastreo SSL en sistemas UNIX, Linux o Windows , consulte strmqtrc.

Referencia relacionada
Mandatos runmqckm y runmqakm

Configuración de un repositorio de claves en los sistemas UNIX, Linux, and Windows

Puede configurar un repositorio de claves utilizando la interfaz de usuario iKeyman o bien los mandatos **iKeycmd** o **runmqakm**.

Acerca de esta tarea

Una conexión SSL o TLS requiere un *repositorio de claves* en cada extremo de la conexión. Cada gestor de colas de IBM WebSphere MQ y IBM WebSphere MQ MQI client debe tener acceso a un repositorio de claves. Para más información, consulte “El repositorio de claves SSL o TLS” en la página 25.

En sistemas UNIX, Linux, and Windows, los certificados digitales se almacenan en un archivo de base de datos de claves que se gestiona mediante la interfaz de usuario **iKeyman** o mediante los mandatos de **iKeycmd** o **runmqakm**. Estos certificados digitales tienen etiquetas. Una etiqueta específica asocia un certificado personal a un gestor de colas o un IBM WebSphere MQ MQI client. SSL y TLS utilizan ese certificado con fines de autenticación. En los sistemas UNIX, Linux, and Windows, IBM WebSphere MQ utiliza `ibmwebspheremq` como prefijo de etiqueta para evitar la confusión con certificados de otros productos. El prefijo va seguido por el nombre del gestor de colas o ID de usuario de inicio de sesión de IBM WebSphere MQ MQI client, cambiado a minúsculas. Asegúrese de especificar la etiqueta completa del certificado en minúsculas.

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 WebSphere MQ . Por ejemplo, C:\program files\IBM\WebSphere 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 WebSphere 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 "Cambio de ubicación del repositorio de claves para un gestor de colas en sistemas UNIX, Linux o Windows" en la página 124. 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 se ejecutan los mandatos **iKeyman** o **iKeycmd** deben tener permiso de escritura para el directorio en el que se crea o se 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 se ejecuta el mandato **iKeyman** o **iKeycmd** predeterminado, debe ser un miembro del grupo mqm. Para un IBM WebSphere MQ MQI client, si ejecuta el mandato **iKeyman** o **iKeycmd** desde un ID de usuario diferente del ID con el que se ejecuta el cliente, debe modificar los permisos de archivo para que el IBM WebSphere MQ MQI client acceda al archivo de base de datos de claves en tiempo de ejecución. Para obtener más información, consulte "Acceso y aseguramiento de los archivos de bases de datos de claves en Windows" en la página 122 o "Acceso y protección de los archivos de base de datos de claves en sistemas UNIX and Linux" en la página 122.

En **iKeyman** o bien **iKeycmd** versión 7.0, las nuevas bases de datos de claves se cumplimenta automáticamente con un conjunto predefinido de certificados de CA (entidad emisora de certificados). En **iKeyman** o **iKeycmd** versión 8.0, las bases de datos de claves no se cumplimentan automáticamente, con lo cual la configuración inicial es más segura porque sólo se incluyen los certificados de CA deseados en el archivo de base de datos de claves.

Nota: Debido a este cambio en el comportamiento de GSKit versión 8.0 que provoca que los certificados de CA ya no se añadan automáticamente al repositorio, el usuario debe añadir manualmente sus 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 en sistemas UNIX, Linux, and Windows con GSKit versión 8.0" en la página 122.

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:
   - En sistemas UNIX, Linux, and Windows:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Utilización de 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 WebSphere MQ, debe ser `cms`.)

-stash
Guarda la contraseña de la base de datos de claves en un archivo.

-fips
Inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC y este componente debe inicializarse correctamente 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 iKeyman 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:\Program Files\IBM\WebSphere 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 de IBM WebSphere MQ: `/var/mqm/ssl` (en sistemas UNIX and Linux) o `C:\mqm\ssl` (en sistemas Windows).

8. Pulse **Abrir**.

   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 SSL o 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 aseguramiento de los archivos de bases de datos de claves en Windows" en la página 122 o "Acceso y protección de los archivos de base de datos de claves en sistemas UNIX and Linux" en la página 122.

Acceso y aseguramiento de los archivos de bases 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.

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-----).

Adición de certificados de CA predeterminados a un repositorio de claves en sistemas UNIX, Linux, and Windows con GSKit versión 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 versión 7.0, el comportamiento al crear un nuevo repositorio de claves era añadir automáticamente un conjunto de certificados de CA para las autoridades emisoras de certificados

utilizadas con mayor frecuencia. 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 iKeyman

Realice los pasos siguientes en la máquina a la que desea añadir el certificado de CA:

1. Inicie la GUI de iKeyman 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 **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 para la organización en cuyos certificados de CA desea confiar para ver la lista completa de certificados CA válidos.
11. Seleccione los certificados de CA en los que desea confiar de la lista y pulse **Aceptar**. Los certificados se añaden al repositorio de claves.

Utilización de la línea de mandatos

Utilice los siguientes mandatos para enumerar y luego añadir certificados de CA mediante el mandato iKeycmd:

- 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 WebSphere MQ confíe en todos los certificados personales firmados por el certificado de CA. Considere cuidadosamente los certificados de CA en los que desea confiar y sólo añada 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.

Ubicación del repositorio de claves para un gestor de colas en sistemas 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 WebSphere 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 sistemas 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 WebSphere MQ .

Cambio de ubicación del repositorio de claves para un gestor de colas en sistemas UNIX, Linux o 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 sistemas 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:\Program Files\IBM\WebSphere MQ\Qmgrs\QM1\ssl\Mykey')
```

El archivo de base de datos de claves tiene el nombre de archivo completo: `C:\Program Files\IBM\WebSphere 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 modificar los atributos del gestor de colas utilizando WebSphere MQ Explorer o 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 sistemas UNIX, Linux, and Windows" en la página 138.

Ubicación del repositorio de claves para un cliente MQI de IBM WebSphere MQ en sistemas 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 cliente MQI de IBM WebSphere MQ . 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 cliente MQI de IBM WebSphere MQ en sistemas UNIX, Linux, and Windows" en la página 125. El valor establecido en una llamada MQCONNX altera temporalmente el valor de MQSSLKEYR.

Especificación de la ubicación del repositorio de claves para un cliente MQI de IBM WebSphere MQ en sistemas UNIX, Linux, and Windows

No hay ningún repositorio de claves predeterminado para un cliente MQI de IBM WebSphere MQ . 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 del cliente MQI de IBM WebSphere MQ de dos maneras:

- Estableciendo la variable de entorno MQSSLKEYR. Por ejemplo, en sistemas 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:\Program Files\IBM\WebSphere MQ\ssl\key
  ```

  El archivo de base de datos de claves tiene el nombre de archivo completo:

  ```
  C:\Program Files\IBM\WebSphere 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.

Cuándo entran en vigor los cambios en los certificados o en el almacén de certificados en sistemas UNIX, Linux o 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 SSL.

- Cuando un nuevo proceso de canal de entrada individual TCP/IP recibe por primera vez una petición para iniciar un canal SSL.
- Cuando se emite el mandato MQSC REFRESH SECURITY TYPE(SSL) para renovar el entorno SSL de Websphere MQ.
- Para los procesos de la aplicación cliente, cuando se cierra la última conexión SSL del proceso. La siguiente conexión SSL 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 SSL. Si el proceso de agrupación de procesos ya ha ejecutado un canal SSL 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 SSL. Si el proceso iniciador de canal ya ha ejecutado un canal SSL 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 petición para iniciar un canal SSL. Si el escucha ya ha ejecutado un canal SSL y desea que el cambio entre en vigor de forma inmediata, ejecute el mandato MQSC REFRESH SECURITY TYPE(SSL).

También puede renovar el entorno SSL de WebSphere MQ utilizando IBM WebSphere MQ Explorer o los mandatos PCF.

Creación de un certificado personal autofirmado en los sistemas UNIX, Linux, and Windows

Puede crear un certificado autofirmado mediante iKeyman, iKeycmd o runmqakm.

Nota: IBM WebSphere 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” en la página 212.

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. WebSphere MQ da soporte a tres tipos diferentes de CipherSpec. Si desea más detalles, consulte “Interoperatividad de Elliptic Curve y CipherSpecs RSA” en la página 37 en el tema “Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ” en la página 36. Para utilizar las CipherSpecs de tipo 1 (aquellas cuyos nombres 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.

Utilización de iKeyman

iKeyman no proporciona ninguna opción compatible con FIPS. Si tiene que gestionar certificados SSL o TLS de una forma que sea compatible con FIPS, utilice el mandato **runmqakm**.

Utilice el siguiente procedimiento para obtener un certificado autofirmado para su gestor de colas o cliente MQI de WebSphere MQ:

1. Inicie la GUI de iKeyman mediante 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** (Sistema de gestión de certificados).
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 en el que desea guardar 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 menú **Crear**, pulse **Nuevo certificado autofirmado**. Se visualiza la ventana Crear nuevo certificado autofirmado.
9. En el campo **Etiqueta de clave**, escriba:
   - Para un gestor de colas, `ibmwebspheremq` seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1` o,
   - Para un cliente de WebSphere MQ, `ibmwebspheremq` seguido del ID de usuario de inicio de sesión en minúsculas; por ejemplo, `ibmwebspheremqmyuserid`.
10. Escriba o seleccione un valor para cualquier campo de **Distinguished name**o cualquiera de los campos de **Subject alternative name** .
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.

Utilización de la línea de mandatos

Utilice los siguientes mandatos para crear un certificado personal autofirmado utilizando iKeycmd o runmqakm:

- Utilizando iKeycmd en sistemas UNIX, Linux y 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_dsname` *`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
```

`-db` *`filename`*	El nombre de archivo completo de una base de datos de claves CMS.
`-pw` *`password`*	La contraseña de la base de datos de claves CMS.
`-label` *`label`*	Etiqueta de clave adjunta al certificado.
`-dn` *`distinguished_name`*	Nombre distinguido X.500 especificado entre comillas dobles. Se requiere al menos un atributo. Puede proporcionar varios atributos OU o DC.
`-size` *`key_size`*	El tamaño de clave. Para iKeycmd, el valor puede ser 512 ó 1024. Para runmqakm, el valor puede ser 512, 1024, 2048 ó 4096.

`-x509version` *version*	Versión del certificado X.509 que se debe crear. El valor puede ser 1, 2 o 3. El valor predeterminado es 3.
`-expire` *days*	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. Esta modalidad inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC 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 **runmqakm** no se ejecuta correctamente.
`-sig_alg`	Para runmqakm, algoritmo de hash utilizado durante la creación de un certificado autofirmado. Este algoritmo hash se utiliza para crear la firma asociada con el certificado autofirmado recién creado. 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`.
`-sig_alg`	Para iKeycmd, algoritmo de signatura asimétrica utilizado para crear la entrada del par de claves. El valor puede ser `MD2_WITH_RSA`, `MD2WithRSA`, `MD5_WITH_RSA` , `MD5WithRSA`, `SHA1WithDSA`, `SHA1WithRSA`, `SHA256_WITH_RSA`, `SHA256WithRSA` , `SHA2WithRSA`, `SHA384_WITH_RSA`, `SHA384WithRSA`, `SHA512_WITH_RSA`, `SHA512WithRSA`, `SHA_WITH_DSA`, `SHA_WITH_RSA` , `SHAWithDSA` o `SHAWithRSA`. El valor predeterminado es `SHA1WithRSA`.
`-san_dnsname` *DNS_names*	Lista delimitada por comas o espacios de los nombres de DN para la entrada que se crea.
`-san_emailaddr` *email_addresses*	Lista delimitada por comas o espacios de las direcciones de correo electrónico para la entrada que se crea.
`-san_ipaddr` *IP_addresses*	Lista delimitada por comas o espacios de las direcciones IP para la entrada que se crea.

distributed *Solicitud de un certificado personal en los sistemas UNIX, Linux, and Windows*

Puede solicitar un certificado personal utilizando la interfaz gráfica de usuario **strmqikm** (iKeyman) o desde la línea de mandatos utilizando los mandatos **runmqckm** o **runmqakm**. 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 iKeyman, o desde la línea de mandatos, con las consideraciones siguientes:

- WebSphere 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. WebSphere MQ da soporte a tres tipos distintos de CipherSpec. Si desea más detalles, consulte "Interoperatividad de Elliptic Curve y CipherSpecs RSA" en la página 37 en el tema "Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ" en la página 36.
- 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.
- Sólo 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 145.

Utilización de la interfaz de usuario iKeyman

Acerca de esta tarea

iKeyman no proporciona ninguna opción compatible con FIPS. Si tiene que gestionar certificados SSL o 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 de iKeyman 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** (Sistema de gestión de certificados).
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 las etiquetas siguientes:
   - Para un gestor de colas, escriba ibmwebspheremq seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para un gestor de colas denominado QM1, especifique ibmwebspheremqqm1.
   - Para un IBM WebSphere MQ MQI client, especifique ibmwebspheremq seguido del ID de usuario de inicio de sesión, todo en minúsculas; por ejemplo, ibmwebspheremqmyuserid .
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 **Escriba el nombre de un archivo donde desee guardar 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 129.

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.

Utilización de la línea de mandatos

Procedimiento

Utilice los mandatos siguientes para solicitar un certificado personal utilizando el mandato **runmqckm** o bien **runmqakm**:

- 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* .

- Utilización de 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.

-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.

-size *tamaño_clave*
Especifica el tamaño de clave. Si utiliza **runmqckm**, el valor puede ser 512 o bien 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. Esta modalidad inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC y este componente debe inicializarse correctamente 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 , SHA1WithRSA, SHA256_WITH_RSA, SHA256WithRSA, SHA2WithRSA, SHA384_WITH_RSA , SHA384WithRSA, SHA512_WITH_RSA , SHA512WithRSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA o 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.

Renovación de un certificado personal existente en sistemas UNIX, Linux, and Windows

Puede renovar un certificado personal utilizando la interfaz de usuario iKeyman o utilizando los mandatos **iKeycmd** o **runmqakm** .

Antes de empezar

Si tiene un requisito de utilizar tamaños de clave mayores para sus certificados personales, los pasos de renovación descritos a continuación no funcionan, porque la solicitud de certificado que se ha vuelto a crear se genera a partir de una clave ya existente.

Siga los pasos descritos en "Solicitud de un certificado personal en los sistemas UNIX, Linux, and Windows" en la página 128 para crear una nueva solicitud de certificado, utilizando los tamaños de clave necesarios. Este proceso sustituye la clave existente.

Acerca de esta tarea

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 iKeyman

Acerca de esta tarea

iKeyman no proporciona ninguna opción compatible con FIPS. Si tiene que gestionar certificados SSL o 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 de iKeyman utilizando el mandato **strmqikm** en sistemas 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** (Sistema de gestión de certificados).
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 132.
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 mediante el mandato **iKeycmd** o bien **runmqakm**:

- Utilización de **iKeycmd** en los 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 sistemas UNIX, Linux y Windows” en la página 133.

Recepción de certificados personales en un repositorio de claves en sistemas UNIX, Linux y 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 iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna 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 cliente MQI de WebSphere MQ reciba un certificado personal en el archivo de base de datos de claves:

1. Inicie la GUI de iKeyman 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

Utilice los mandatos siguientes para añadir un certificado personal a un archivo de bases de datos de claves utilizando iKeycmd :

- En UNIX, Linux y Windows, emita el mandato siguiente:

```
runmqckm -cert -receive -file filename -db filename -pw
password
        -format ascii
```

donde:

`-file` *filename*	es el nombre de archivo completo del archivo que contiene el certificado personal.
`-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.
`-format` *ascii*	es el formato del certificado. El valor puede ser `ascii` para ASCII Base64-encoded o `binary` para datos DER binarios. El valor predeterminado es `ascii`.

Si utiliza hardware de cifrado, consulte el apartado "Importación de un certificado personal a su hardware PKCS #11" en la página 147.

Extracción de un certificado de CA desde un repositorio de claves

Siga este procedimiento para extraer un certificado de CA.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina desde la que desea extraer el certificado de CA:

1. Inicie la GUI de iKeyman mediante 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 CA utilizando iKeycmd :

- En UNIX, Linux y 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`.

Extracción de la parte pública de un certificado autofirmado de un repositorio de claves en sistemas UNIX, Linux y Windows

Siga este procedimiento para extraer la parte pública de un certificado autofirmado.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna 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 de iKeyman utilizando el mandato **strmqikm** (en 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 del que desea extraer 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.
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 siguientes mandatos para extraer la parte pública de un certificado autofirmado utilizando iKeycmd o runmqakm:

- En UNIX, Linux y 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`.

Adición de un certificado de CA (o la parte pública de un certificado autofirmado) a un repositorio de claves, en sistemas 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: 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 asegurarse de que el certificado tenga una codificación ASCII (UTF-8) o binaria (DER), porque IBM Global Secure Toolkit (GSKit) no admite certificados con otros tipos de certificación. Debe añadir los certificados en orden estrictamente descendente, empezando por el raíz, seguido del certificado de CA inmediatamente debajo de éste en la cadena.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna opción compatible con FIPS.

Realice los pasos siguientes en la máquina a la que desea añadir el certificado de CA:

1. Inicie la GUI de iKeyman 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 **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 **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

Utilice los mandatos siguientes para añadir un certificado CA utilizando iKeycmd :

- En UNIX, Linux y Windows, emita el mandato siguiente:

```
runmqckm -cert -add -db filename -pw password -label label -file filename
        -format ascii
```

donde:

-db *filename*	es el nombre de vía de acceso completo de la base de datos de claves de CMS.
-pw *password*	es la contraseña de la base de datos de claves CMS.
-label *label*	es la etiqueta adherida al certificado.
-file *filename*	es el nombre del archivo que contiene el certificado.
-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`.

Exportación de un certificado personal desde un repositorio de claves

Siga este procedimiento para exportar un certificado personal.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina desde la que desea exportar el certificado personal:

1. Inicie la GUI de iKeyman 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 siguientes mandatos para exportar un certificado personal utilizando iKeycmd:

- En UNIX, Linux y 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.
-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.

Importación de un certificado personal en un repositorio de claves en sistemas 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 sistemas UNIX, Linux, and Windows" en la página 136).

Los archivos de PKCS #12 deben considerarse temporales y deben suprimirse después de su uso.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna opción compatible con FIPS.

Efectúe los pasos siguientes en la máquina a la que desea importar el certificado personal:

1. Inicie la GUI de iKeyman mediante 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. Esta ventana permite 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. Esto se puede utilizar para cambiar la etiqueta del certificado personal a la necesaria para WebSphere MQ con tal de asociar el certificado con el gestor de colas o el cliente concreto (`ibmwebspheremqqm1` por ejemplo).
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 mediante iKeycmd, utilice los mandatos siguientes:

- En UNIX, Linux y 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`.

iKeycmd no proporciona un mandato para cambiar las etiquetas de certificados directamente. Utilice los pasos siguientes para cambiar una etiqueta de certificado:

1. Exporte el certificado a un archivo PKCS #12 utilizando 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.

Importación desde un archivo .pfx Microsoft

Siga este procedimiento para importar desde un archivo .pfx de Microsoft utilizando iKeyman. No puede utilizar runmqakm para importar un archivo .pfx.

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 de iKeyman utilizando el mandato **strmqikm** (en Linux, UNIX o Windows). 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 nombre descriptivo debe tener el formato `ibmwebspheremq`, seguido por el nombre del gestor de colas o el ID de inicio de sesión del cliente MQI de WebSphere MQ en minúsculas.
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 iKeycmd

Importación desde un archivo PKCS #7

Las herramientas iKeyman e iKeycmd no dan soporte a los archivos PKCS #7 (.p7b). Utilice la herramienta runmqckm para importar certificados desde un archivo PKCS #7.

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.

Supresión de un certificado de un repositorio de claves en sistemas UNIX, Linux, and Windows

Utilice este procedimiento para suprimir certificados personales o de CA.

Utilización de iKeyman

Si tiene que gestionar certificados SSL de una forma que sea compatible con el estándar FIPS, utilice el mandato runmqakm. iKeyman no proporciona ninguna opción compatible con FIPS.

1. Inicie la GUI de iKeyman 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 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 desde un repositorio de claves" en la página 137).
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 siguientes mandatos para suprimir un certificado utilizando iKeycmd o runmqakm:

- En UNIX, Linux y 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. Esta modalidad inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC 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 **runmqakm** no se ejecuta correctamente.

Generación de contraseñas fuertes para la protección de repositorios de claves

Puede generar contraseñas fuertes para la protección de repositorios de claves utilizando el mandato **runmqakm**.

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 iKeyman, no es necesario incluir entre comillas ni escapar la contraseña. No es necesario porque el shell del sistema operativo no afecta a la entrada de entrada datos en estos casos.

Configuración del hardware criptográfico en sistemas 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 sistemas UNIX, Linux o Windows utilizando cualquiera de los métodos siguientes:

- Utilice el mandato ALTER QMGR MQSC con el parámetro SSLCRYP, tal como se describe en ALTER QMGR.
- Utilice IBM WebSphere MQ Explorer para configurar el 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 WebSphere MQ en sistemas UNIX, Linux o Windows utilizando cualquiera 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_PCS11 del parámetro SSLCRYP, la etiqueta de la señal PKCS #11 debe especificarse enteramente en minúsculas.
- 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. Esto se describe en “Gestión de certificados en el hardware PKCS #11” en la página 143.

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 WebSphere 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:

   - En sistemas UNIX, Linux, and Windows:

     ```
     runmqckm -keydb -create -db filename -pw password -type cms -stash
     ```

   - Utilización de 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 WebSphere MQ, debe ser `cms`.)

   -stash
   Guarda la contraseña de la base de datos de claves en un archivo.

   -fips
   Inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC y este componente debe inicializarse correctamente 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 iKeyman ejecutando el mandato **strmqikm**.
4. Pulse **Archivo de base de datos de claves** > **Abrir**.
5. Pulse **Tipo de base de datos de claves** y seleccione **PKCS11Direct**.
6. 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 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 Windows y Linux x86 de 32 bits son las únicas excepciones, ya que los programas iKeyman e iKeycmd son de 32 bits en esas plataformas.

7. 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.
8. En el campo **Contraseña de señal de cifrado**, escriba la contraseña que estableció al configurar el hardware de cifrado.
9. 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 "13" en la página 145.

   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**.
10. 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`.
11. En el campo **Ubicación**, escriba la vía de acceso, por ejemplo:
    - Para un gestor de colas: `/var/mqm/qmgrs/QM1/ssl`
    - Para un cliente IBM WebSphere MQ MQI: `/var/mqm/ssl`

    Pulse **Aceptar**. Se abre la ventana Solicitud de contraseña.
12. Escriba una contraseña.

    Si ha seleccionado **Abrir archivo de base de datos de claves secundarias existente** en el paso "9" en la página 145, escriba una contraseña en el campo **Contraseña**.

    Si ha seleccionado **Crear nuevo archivo de base de datos de claves secundarias** en el paso "9" en la página 145, 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 SSL 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).
13. Pulse **Aceptar**. Se visualiza la sección de contenido de la base de datos de claves.

Solicitud de un certificado personal para el hardware PKCS #11
Siga este procedimiento para un gestor de colas o un cliente MQI de IBM WebSphere MQ para solicitar un certificado personal para el hardware de cifrado.

Utilización de la interfaz de usuario iKeyman

Acerca de esta tarea

Nota: WebSphere MQ no admite 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 de la interfaz de usuario iKeyman, efectúe 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 143.
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 las etiquetas siguientes:
   - Para un gestor de colas, escriba `ibmwebspheremq` seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para un gestor de colas denominado `QM1`, especifique `ibmwebspheremqqm1`.
   - Para un IBM WebSphere MQ MQI client, especifique `ibmwebspheremq` seguido del ID de usuario de inicio de sesión, todo en minúsculas; por ejemplo, `ibmwebspheremqmyuserid` .
4. 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**. Si desea más información sobre estos campos, consulte "Nombres distinguidos" en la página 11.
5. En el campo **Escriba el nombre de un archivo donde desee guardar la solicitud de certificado**, acepte el valor predeterminado `certreq.arm` o escriba un valor nuevo con una vía de acceso completa.
6. Pulse **Aceptar**.

   Se abrirá una ventana de confirmación.
7. 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 "5" en la página 146.
8. 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**:

- 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* .

- Utilización de 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.

-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.

-size *tamaño_clave*
Especifica el tamaño de clave. Si utiliza **runmqckm**, el valor puede ser 512 o bien 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. Esta modalidad inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC y este componente debe inicializarse correctamente 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 , SHA1WithRSA, SHA256_WITH_RSA, SHA256WithRSA, SHA2WithRSA, SHA384_WITH_RSA , SHA384WithRSA, SHA512_WITH_RSA , SHA512WithRSA, SHA_WITH_DSA, SHA_WITH_RSA, SHAWithDSA o 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.

Importación de un certificado personal a su hardware PKCS #11
Siga este procedimiento para un gestor de colas o un cliente MQI de IBM WebSphere MQ para importar un certificado personal al hardware de cifrado.

Utilización de iKeyman

Procedimiento

Para solicitar un certificado personal de la interfaz de usuario iKeyman, efectúe 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 143.
2. Pulse **Recibir**. Se abre la ventana Recibir certificado de un archivo.
3. Seleccione el **Tipo de datos** del nuevo certificado personal; por ejemplo, `Base64-encoded ASCII data` para un archivo con la extensión `.arm`.
4. Escriba el nombre de archivo de certificado y la ubicación del certificado personal nuevo, o pulse **Examinar** para seleccionar el nombre y la ubicación.
5. 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.
6. Pulse **Sí** o **No**. Se abre la ventana Entrar una etiqueta.
7. Escriba una etiqueta.

   Por ejemplo, podría utilizar la misma etiqueta que la que empleó cuando solicitó el certificado personal. Tenga en cuenta que la etiqueta debe tener el formato de IBM WebSphere MQ correcto:

   - Para un gestor de colas, `ibmwebspheremq` seguido por el nombre del gestor de colas en minúsculas. Por ejemplo, para un gestor de colas denominado QM1, la etiqueta sería: `ibmwebspheremqqm1`.
   - En el caso de un cliente MQI de IBM WebSphere MQ, `ibmwebspheremq` seguido por su ID de usuario de inicio de sesión en minúsculas. Por ejemplo, para un ID de usuario `MyUserID`, la etiqueta sería:`ibmwebspheremqmyuserid`.
8. 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.

Utilización de la línea de mandatos

Procedimiento

Para solicitar un certificado personal desde una línea de mandatos, efectúe los pasos siguientes:

1. Abra una ventana de mandatos configurada para su entorno.
2. Escriba el mandato apropiado para el sistema operativo y para la configuración:

   - En sistemas Windows, UNIX and Linux, utilice uno de los mandatos siguientes:

     ```
     runmqckm -cert -receive -file filename -crypto path
     -tokenlabel hardware_token -pw hardware_password -format cert_format
     ```

     ```
     runmqakm -cert -receive -file filename -crypto path
     -tokenlabel hardware_token -pw hardware_password -format cert_format -fips
     ```

   donde:

   -file *nombrearchivo*
   Especifica el nombre de archivo completo del archivo que contiene el certificado personal.

   -crypto *vía_acceso*
   Especifica la vía de acceso completa a la biblioteca PKCS #11 suministrada con el hardware.

   -tokenlabel *señal_hardware*
   Especifica la etiqueta suministrada a la parte de almacenamiento del hardware de cifrado durante la instalación.

-pw *contraseña_hardware*
Especifica la contraseña para el acceso al hardware.

-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. Esta modalidad inhabilita el uso de la biblioteca de cifrado BSafe. Sólo se utiliza el componente ICC y este componente debe inicializarse correctamente 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.

Identificación y autenticación de usuarios

Puede identificar y autenticar usuarios utilizando la estructura MQCSP o en varios tipos de programa de salida de usuario.

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.

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 WebSphere MQ. En esta situación, la autenticación mutua tiene lugar entre dos gestores de colas, o entre un gestor de colas y el usuario de una aplicación cliente MQI de WebSphere MQ.

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 la página 106.

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.

SSL (Secure Sockets Layer) y TLS (Transport Layer Security) utilizan técnicas PKI como las que se acaban de describir. Para obtener más información sobre cómo SSL y TLS realizan la autenticación, consulte "Conceptos SSL (Secure Sockets Layer) y 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 231. 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 77.

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 153.

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.

Usuarios privilegiados

Un usuario privilegiado es aquel que tiene autorización administrativa completa para WebSphere MQ.

Además de los usuarios listados en la tabla siguiente, los miembros de cualquier grupo con autorización `+crt` para colas son indirectamente administradores. De forma similar, cualquier usuario que tenga autorización `+set` en el gestor de colas y autorización `+put` en la cola de mandatos es un administrador.

No otorgue estos privilegios a usuarios y aplicaciones ordinarios.

Tabla 13. Usuarios privilegiados por plataforma.

Una tabla de usuarios privilegiados. En Windows, SYSTEM, todos los miembros del grupo mqm y todos los miembros del grupo Administradores son usuarios privilegiados. En los sistemas UNIX and Linux, todos los miembros del grupo mqm son usuarios privilegiados. En IBM i, los perfiles (usuarios) qmqm y qmqmadm, todos los miembros del grupo qmqmadm y cualquier usuario definido con el valor *ALLOBJ son usuarios privilegiados.

Plataforma	**Usuarios privilegiados**
Sistemas Windows	• SISTEMA • Miembros del grupo mqm • Miembros del grupo Administradores
Sistemas UNIX and Linux	• Miembros del grupo mqm

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.

El componente del servicio de autorización proporcionado con IBM WebSphere MQ se llama Gestor de autoridad de objeto (OAM). El OAM autoriza a los usuarios basándose en el ID contenido en el MQCSP, pero no valida la contraseña. Puede implementar la validación de contraseña en el servicio de autorización utilizando las salidas encadenadas con el OAM, o sustituyendo el OAM por un servicio de autorización alternativo.

Puede alterar el MQCSP en una salida de seguridad.

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 cliente MQI de WebSphere MQ. En esta situación, la autenticación mutua tiene lugar entre dos gestores de colas, o entre un gestor de colas y el usuario de una aplicación cliente MQI de WebSphere MQ.

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 la página 106.

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.

SSL (Secure Sockets Layer) y TLS (Transport Layer Security) utilizan técnicas PKI como las que se acaban de describir. Para obtener más información sobre cómo SSL lleva a cabo la autenticación, consulte "Conceptos SSL (Secure Sockets Layer) y 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 231. 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 77.

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 153.

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 SSL, 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 sistemas UNIX, Linux y Windows , el soporte SSL de WebSphere MQ 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. Las clases IBM WebSphere MQ classes for Java y las clases IBM WebSphere MQ classes for JMS no pueden utilizar la información de OCSP en un archivo de tabla de definición de canal de cliente. Sin embargo, puede configurar OCSP como se describe en la sección Utilización del protocolo de certificados en línea.

En z/Os y IBM i WebSphere MQ , el soporte SSL comprueba si hay certificados revocados utilizando CRL y ARL sólo en servidores LDAP.

Para obtener más información sobre entidades emisoras de

certificados, consulte “Certificados digitales” en la página 9.

Certificados revocados y OCSP

IBM WebSphere 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 solamente se aplica a WebSphere MQ en sistemas Windows y UNIX and Linux.

Para comprobar el estado de revocación de un certificado digital utilizando OCSP, WebSphere MQ puede utilizar dos métodos para determinar el respondedor 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 o SSL, 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 WebSphere MQ no puede acceder al programa de respuesta OCSP.
- El respondedor OCSP ha enviado una respuesta, pero WebSphere 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 WebSphere MQ recibe una salida OCSP `Desconocido`, su comportamiento depende del valor del atributo OCSPAuthentication. Para gestores de colas, este atributo se almacena en la stanza SSL del archivo `qm.ini` para sistemas UNIX and Linux, o en el registro de Windows. Puede establecerse utilizando IBM WebSphere MQ Explorer. Para clientes, se mantiene en la stanza SSL del archivo de configuración cliente.

Si se recibe un resultado `Desconocido` y se ha definido OCSPAuthentication en REQUIRED (el valor predeterminado), WebSphere MQ rechazará 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 un resultado `Desconocido` y se ha definido OCSPAuthentication en OPTIONAL, WebSphere MQ permitirá que se inicie el canal y no se generarán avisos ni mensajes de sucesos SSL.

Si se recibe una salida `Desconocido` y OCSPAuthentication está establecido en WARN, se inicia el canal SSL pero IBM WebSphere 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 necesita configurar ningún certificado adicional; los pasos que ya ha tomado para establecer la conectividad SSL son 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 ese certificado está firmado por una CA que ya es fiable para la conectividad SSL) 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 sistemas UNIX, Linux, and Windows” en la página 136. Cuando se añade un certificado CA, de forma predeterminada se añade como raíz fiable, que es el valor necesario en este contexto. Si no se añade este certificado, WebSphere MQ no podrá verificar la firma digital en la respuesta OCSP y la comprobación OCSP generará un resultado Desconocido, que puede hacer que IBM WebSphere MQ cierre el canal, dependiendo del valor de OCSPAuthentication.

OCSP (Online Certificate Status Protocol) en aplicaciones cliente de JMS y Java.

Debido a una limitación de la API de Java, WebSphere MQ puede utilizar la comprobación de revocación de certificados OCSP (Online Certificate Status Protocol) para sockets seguros SSL y 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 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

Nombre de propiedad	Descripción
	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 SSL o 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 los arreglos temporales y el mantenimiento del producto 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 utiliza 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 WebSphere 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.

WebSphere 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 MQI de WebSphere MQ recibe un certificado, comprueba la CRL para confirmar que el certificado sigue siendo válido. WebSphere MQ comprueba en primer lugar la memoria caché, si ésta existe. Si la CRL no está en la memoria caché, WebSphere MQ examina las ubicaciones del servidor de CRL LDAP en el orden en que se producen en la lista de nombres de objetos de información de autenticación especificada mediante el atributo *SSLCRLNamelist*, hasta que WebSphere MQ encuentra 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.

Para obtener más información sobre LDAP, consulte Utilización de servicios de Lightweight Directory Access Protocol con WebSphere MQ para Windows.

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 WebSphere MQ SSL, 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.

La Figura 12 en la página 159 muestra un archivo LDIF de ejemplo que puede crear como entrada para el servidor LDAP para cargar las CRL y las ARL emitidas por la CA1, que es una Entidad emisora de certificados ficticia con el nombre distinguido "CN=CA1, OU=Test, O=IBM, C=GB", configurada por una organización de prueba de IBM.

```
dn: o=IBM, c=GB
o: IBM
objectclass: top
objectclass: organization

dn: ou=Test, o=IBM, c=GB
ou: Test
objectclass: organizationalUnit

dn: cn=CA1, ou=Test, o=IBM, c=GB
cn: CA1
objectclass: cRLDistributionPoint
objectclass: certificateAuthority
authorityRevocationList;binary:: (DER format data)
certificateRevocationList;binary:: (DER format data)
caCertificate;binary:: (DER format data)
```

Figura 12. Archivo LDIF de ejemplo para una Entidad emisora de certificados. Puede variar de implementación en implementación.

La Figura 13 en la página 160 muestra la estructura DIT que el servidor LDAP crea cuando carga el archivo LDIF de ejemplo que se muestra en la Figura 12 en la página 159 junto con un archivo similar para la CA2, una Entidad emisora de certificados ficticia establecida por la organización PKI, también dentro de IBM.

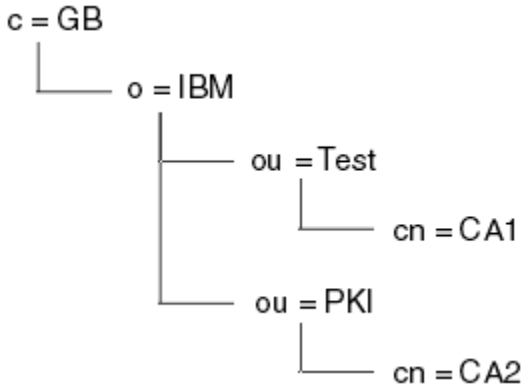


Figura 13. Ejemplo de una estructura de árbol de la información de directorios LDAP

WebSphere MQ 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 160.

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.

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 guardan en una lista de nombres, que está especificada en el atributo del gestor de colas *SSLCRLNamelist*.

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.

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.

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.
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 *SSLCRLNamelist*. El valor inicial del gestor de colas para este atributo está en blanco.

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.

Acceso a las CRL y las ARL utilizando IBM WebSphere MQ Explorer
Puede utilizar IBM WebSphere 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 con el botón derecho del ratón en la carpeta **Lista 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 161).
   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 161) en el campo **Nombre de la lista de CRL**.

Acceso a las CRL y las ARL con un cliente MQI de IBM WebSphere MQ

Tiene tres opciones para especificar los servidores LDAP que contienen CRL para la comprobación por parte de un cliente MQI de IBM WebSphere MQ .

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 WebSphere MQ que se ejecuta en Linux (plataforma zSeries).

Ubicación de un programa de respuestas OCSP, y de servidores LDAP que contienen CRL
En un sistema cliente IBM WebSphere MQ MQI, puede especificar la ubicación de un programa de respuesta OCSP y de los servidores LDAP (Lightweight Directory Access Protocol) que contienen listas de revocación de certificados (CRL).

Puede especificar estas ubicaciones de tres modos, mostrados aquí en orden de mayor a menor precedencia.

Cuando una aplicación cliente MQI de WebSphere MQ 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 cliente MQI de WebSphere MQ 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 cliente MQI de WebSphere MQ 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.

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 WebSphere MQ 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:

En las plataformas de servidor AIX, HP-UX, Linux, 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, **SSLCRLNameList**, 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 cliente MQI de WebSphere MQ 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 del gestor de colas, **SSLCRLNameList**, en el nombre de la lista de nombres, la ubicación de la revocación del certificado se copia en la tabla de definiciones de canal de cliente asociada al 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 cliente MQI de WebSphere MQ 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 del gestor de colas, **SSLCRLNameList**, se establece en blanco, la ubicación de revocación del 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.

Utilización de Active Directory en 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 CRL y ARL con clases IBM WebSphere MQ para Java y clases IBM WebSphere MQ para JMS

Las clases IBM WebSphere MQ para Java y las clases IBM WebSphere MQ para las CRL de acceso JMS son diferentes de otras plataformas.

Para obtener información sobre cómo trabajar con CRL y ARL con clases IBM WebSphere MQ para Java, consulte Utilización de listas de revocación de certificados.

Para obtener información sobre cómo trabajar con CRL y ARL con clases IBM WebSphere MQ para 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 WebSphere MQ Explorer.

Los mandatos MQSC siguientes actúan en los objetos de información de autenticación:

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

Para obtener una descripción completa de estos mandatos, consulte Mandatos de script (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

Para obtener una descripción completa de estos mandatos, consulte Definiciones de los formatos de mandatos programables .

En plataformas donde esté disponible, también puede utilizar WebSphere MQ Explorer.

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.

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. También contiene una lista de comprobación que puede utilizarse para determinar qué tareas deben realizarse para aplicar la seguridad al sistema y consideraciones para otorgar a los usuarios la autorización para administrar IBM WebSphere MQ y trabajar con objetos de IBM WebSphere MQ. Si los mecanismos de seguridad proporcionados no satisfacen sus necesidades, puede desarrollar sus propios programas de salida de canal.

Control del acceso a los objetos mediante el OAM en sistemas UNIX, Linux y Windows

El Gestor de autorizaciones sobre objetos (OAM) proporciona una interfaz de mandatos para otorgar y revocar autorización a objetos WebSphere MQ.

Debe tener la autorización adecuada para utilizar estos mandatos, como se describe en “Autorización para administrar IBM WebSphere MQ en sistemas UNIX, Linux, and Windows” en la página 202. Los ID de usuario que están autorizados para administrar WebSphere 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.

Otorgar acceso a un objeto IBM WebSphere MQ en sistemas UNIX, Linux, and Windows

Utilice el mandato de control **setmqaut** o el mandato PCF **MQCMD_SET_AUTH_REC** para proporcionar a los usuarios y grupos de usuarios acceso a los objetos de IBM WebSphere MQ .

Para obtener una definición completa del mandato de control **setmqaut** y su sintaxis, consulte setmqaut, y para 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 de OAM en sistemas UNIX, Linux, and Windows" en la página 166.
- 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 utilización del 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 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.

Utilización de perfiles genéricos de OAM en sistemas UNIX, Linux, and Windows

Los perfiles genéricos de OAM le permiten establecer la autorización que un usuario tiene sobre muchos objetos a la vez, en lugar de tener que emitir mandatos **setmqaut** separados para cada objeto individual cuando se crea.

Utilizando perfiles genéricos en el mandato **setmqaut** puede 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 los procesos, utilice ** como el nombre de perfil para 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.

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 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. **

Volcado de valores de perfil

Para obtener una definición completa del mandato de control **dmpmqaut** y su sintaxis, consulte dmpmqaut, y para obtener una definición completa del mandato 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 and Linux pueden utilizar la opción `-p` para el mandato **dmpmqaut**, deben utilizar `-g groupname` cuando se definen 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: Sólo para WebSphere MQ para 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
```

Utilización de caracteres comodín en perfiles del OAM

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 los procesos, utilice ** como el nombre de perfil para 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.

Prioridades de perfiles

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 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. **

Volcado de valores de perfil

Utilice el mandato de control **dmpmqaut** o el mandato PCF **MQCMD_INQUIRE_AUTH_RECS** para volcar las autorizaciones actuales asociadas a un perfil especificado.

Para obtener una definición completa del mandato de control **dmpmqaut** y su sintaxis, consulte dmpmqaut, y para obtener una definición completa del mandato 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: Sólo para WebSphere MQ para 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
```

Visualización de los valores de acceso

Utilice el mandato de control **dspmqaut**, o el mandato PCF **MQCMD_INQUIRE_ENTITY_AUTH** para ver las autorizaciones que un principal o grupo específico tienen sobre un objeto determinado.

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, y para una definición completa del mandato PCF **MQCMD_INQUIRE_ENTITY_AUTH** y su sintaxis, consulte Consultar autorización de entidad.

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
```

Modificación y revocación del acceso a un objeto IBM WebSphere MQ

Para modificar el nivel de acceso que tiene un usuario o grupo a un objeto, utilice el mandato **setmqaut**. Para revocar el acceso de un usuario determinado que es miembro de un grupo que posee autorización, elimine el usuario del grupo.

El proceso de eliminación de un usuario de un grupo se describe en:

- “Creación y gestión de grupos en Windows” en la página 85
- “Creación y gestión de grupos en HP-UX” en la página 87
- “Creación y gestión de grupos en AIX” en la página 89
- “Creación y gestión de grupos en Solaris” en la página 90
- “Creación y gestión de grupos en Linux” en la página 91

.

Al ID de usuario que crea un objeto IBM WebSphere 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, y para obtener una definición completa del mandato PCF **MQCMD_INQUIRE_ENTITY_AUTH** y su sintaxis, consulte Inquire Entity Authority.

En Windows, suprima las entradas OAM correspondientes a una cuenta de usuario Windows específica antes de suprimir el perfil de usuario. No se pueden eliminar las entradas OAM después de eliminar la cuenta de usuario.

Impedir comprobaciones de acceso de seguridad en los sistemas UNIX, Linux, and Windows

Para desactivar toda 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 (si lo hace, posteriormente no podrá añadir un OAM):

  Consulte Variables de entorno para obtener más información sobre las implicaciones de establecer la variable MQSNOAUT.
- Edite el archivo de configuración del gestor de colas para eliminar el servicio. (Si lo hace, no puede añadir un OAM posteriormente.)

Si utiliza setmqaut o dspmqaut mientras el OAM está inhabilitado, tenga en cuenta las siguientes cuestiones:

- El OAM no valida 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 lugar en el momento de la creación del objeto. Para utilizar WebSphere MQ OAM de nuevo después de que se haya eliminado, es necesario volver a crear el gestor de colas.

Conceptos relacionados

Servicios instalables

Otorgar el acceso necesario a los recursos

Utilice este tema para determinar qué tareas deben realizarse para aplicar la seguridad a su sistema WebSphere MQ.

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 WebSphere 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 174.
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 179.
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 183.
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 184.
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 185
   b) Sí: Consulte “Cómo permitir que las aplicaciones de usuario se conecten con su gestor de colas” en la página 186.

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 14. 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 174
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 175
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 175
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 176
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 177
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 177
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 178

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName ReqdAction
```

- Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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:

– En sistemas UNIX, Linux y Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +dlt, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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.

Procedimiento
- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName ReqdAction
```

- 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:

– En sistemas UNIX, Linux y Windows , cualquier combinación de las autorizaciones siguientes: + chg, + clr, + crt, + dlt, + dsp. + ctrl. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName ReqdAction
```

- Los nombres de las variables tienen los significados siguientes:

  QMgrName
  Nombre del gestor de colas. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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:

  – En sistemas UNIX, Linux y Windows , cualquier combinación de las autorizaciones siguientes: + chg, + clr, + crt, + dlt, + dsp. + ctrl, + ctrlx. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t qmgr -g GroupName ReqdAction
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*MQM) USER(GroupName) AUT(ReqdAction)
MQMNAME('QMgrName')
```

Resultados

Para determinar qué mandatos MQSC puede realizar el usuario 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:

- En sistemas UNIX, Linux y 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.

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName ReqdAction
```

- 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:

  – En sistemas UNIX, Linux y Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName ReqdAction
```

- 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:

– En sistemas UNIX, Linux y Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +ctrl, +ctrlx, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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.

Nota: Los objetos de servicio no existen en z/OS.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName ReqdAction
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*SVC) USER(GroupName) AUT(ReqdAction)
MQMNAME('QMgrName')
```

Resultados

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:

- En sistemas UNIX, Linux y Windows, cualquier combinación de las siguientes autorizaciones: +chg, +clr, +crt, +dlt, +ctrl, +ctrlx, +dsp. La autorización +alladm es equivalente a +chg +clr +dlt +dsp.

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 15. 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 179
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 180
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 180
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 181
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 182
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 182
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 183

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t queue -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*Q) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t topic -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*TOPIC) USER(GroupName) AUT(ALLADM)
MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t channel -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*CHL) USER(GroupName) AUT(ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -t qmgr -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*MQM) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t process -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*PRC) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t namelist -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*NMLIST) USER(GroupName) AUT(*ALLADM)
MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -n ObjectProfile -t service -g GroupName +alladm
```

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('ObjectProfile') OBJTYPE(*SVC) USER(GroupName) AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Utilización del asistente:
  a) En el panel del navegador de WebSphere MQ Explorer, pulse con el botón derecho en el gestor de colas y pulse **Autorizaciones de objetos** > **Añadir autorizaciones basadas en funciones**

  Se abre el asistente Añadir autorizaciones basadas en funciones.
- Para los sistemas UNIX y Windows emita los mandatos siguientes:

```
setmqaut -m QMgrName -n ** -t queue -g GroupName +browse +dsp
setmqaut -m QMgrName -n SYSTEM.ADMIN.COMMAND.QUEUE -t queue -g GroupName +dsp +inq +put
setmqaut -m QMgrName -n SYSTEM.MQEXPLORER.REPLY.MODEL -t queue -g GroupName +dsp +inq +get
setmqaut -m QMgrName -n ** -t topic -g GroupName +dsp
setmqaut -m QMgrName -n ** -t channel -g GroupName +dsp +inq
setmqaut -m QMgrName -n ** -t clntconn -g GroupName +dsp
setmqaut -m QMgrName -n ** -t authinfo -g GroupName +dsp
setmqaut -m QMgrName -n ** -t listener -g GroupName +dsp
setmqaut -m QMgrName -n ** -t namelist -g GroupName +dsp
setmqaut -m QMgrName -n ** -t process -g GroupName +dsp
setmqaut -m QMgrName -n ** -t service -g GroupName +dsp
setmqaut -m QMgrName -t qmgr -g GroupName +dsp +inq +connect
```

Las autorizaciones específicas para SYSTEM.ADMIN.COMMAND.QUEUE y SYSTEM.MQEXPLORER.REPLY.MODEL son necesarias sólo si se desea utilizar MQ Explorer.

- 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')
```

- Para z/OS, emita los siguientes mandatos:

```
RDEFINE MQQUEUE QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQQUEUE) ID(GroupName) ACCESS(READ)
RDEFINE MQTOPIC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQTOPIC) ID(GroupName) ACCESS(READ)
RDEFINE MQPROC QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQPROC) ID(GroupName) ACCESS(READ)
RDEFINE MQNLIST QMgrName.** UACC(NONE)
PERMIT QMgrName.** CLASS(MQNLIST) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.BATCH UACC(NONE)
PERMIT QMgrName.BATCH CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CICS UACC(NONE)
PERMIT QMgrName.CICS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.IMS UACC(NONE)
PERMIT QMgrName.IMS CLASS(MQCONN) ID(GroupName) ACCESS(READ)
RDEFINE MQCONN QMgrName.CHIN UACC(NONE)
PERMIT QMgrName.CHIN CLASS(MQCONN) ID(GroupName) ACCESS(READ)
```

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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

Utilice el asistente para añadir autorizaciones basadas en funciones o los mandatos correspondientes para su sistema operativo.

Procedimiento

- Utilización del asistente:

a) En el panel del navegador de WebSphere MQ Explorer, pulse con el botón derecho en el gestor de colas y pulse **Autorizaciones de objetos** > **Añadir autorizaciones basadas en funciones**

Se abre el asistente Añadir autorizaciones basadas en funciones.

- 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.QUEUE -t queue -g GroupName +dsp +inq +get
setmqaut -m QMgrName -n '**' -t topic -g GroupName +alladm
setmqaut -m QMgrName -n @class -t topic -g GroupName +crt
setmqaut -m QMgrName -n '**' -t channel -g GroupName +alladm
setmqaut -m QMgrName -n @class -t channel -g GroupName +crt
setmqaut -m QMgrName -n '**' -t clntconn -g GroupName +alladm
setmqaut -m QMgrName -n @class -t clntconn -g GroupName +crt
setmqaut -m QMgrName -n '**' -t authinfo -g GroupName +alladm
setmqaut -m QMgrName -n @class -t authinfo -g GroupName +crt
setmqaut -m QMgrName -n '**' -t listener -g GroupName +alladm
setmqaut -m QMgrName -n @class -t listener -g GroupName +crt
setmqaut -m QMgrName -n '**' -t namelist -g GroupName +alladm
setmqaut -m QMgrName -n @class -t namelist -g GroupName +crt
setmqaut -m QMgrName -n '**' -t process -g GroupName +alladm
setmqaut -m QMgrName -n @class -t process -g GroupName +crt
setmqaut -m QMgrName -n '**' -t service -g GroupName +alladm
setmqaut -m QMgrName -n @class -t service -g GroupName +crt
setmqaut -m QMgrName -t qmgr -g GroupName +alladm +conn
```

- Para sistemas Windows, emita los mismos mandatos que para los sistemas UNIX and Linux , pero utilizando el nombre de perfil @CLASS en lugar de @class.
- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ(*ALL) OBJTYPE(*ALL) USER('GroupName') AUT(*ALLADM) MQMNAME('QMgrName')
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

```
setmqaut -m QMgrName -t qmgr -g GroupName -connect
```

- Para IBM i, emita el mandato siguiente:

```
RVKMQMAUT OBJ ('QMgrName') OBJTYPE(*MQM) USER(*ALL) AUT(*CONNECT)
```

- Para z/OS, emita los siguientes mandatos:

```
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. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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 193.

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 186.

En ambos casos, establezca la seguridad de la conexión tal como se describe en “Configurar la seguridad de conexión” en la página 193

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 194.
Tiene aplicaciones que utilizan temas	Consulte “Control del acceso de los usuarios a los temas” en la página 199.
Tiene aplicaciones que consultan en el objeto del gestor de colas	Consulte “Otorgar autorización para consultar en un gestor de colas” en la página 200.
Tiene aplicaciones que utilizan objetos de procesos	Consulte “Otorgar autorización para acceder a procesos” en la página 201.
Tiene aplicaciones que utilizan listas de nombres	Consulte “Otorgar autorización para acceder a listas de nombres” en la página 201.

Cómo proteger la conectividad remota con el gestor de colas

Puede proteger la conectividad remota con el gestor de colas utilizando SSL o 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 SSL o 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 SSL o 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 SSL que reciba en los campos SSLPeerNamePtr y SSLPeerNameLength que se pasan a la salida en la estructura MQCD.
3. Utilizando SSL o 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 SSL o 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) - Utilice ADDRESSMAP si desea proporcionar una sola dirección o dirección comodín. 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 187

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 151. 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 que se muestran a continuación, 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. Para obtener más información sobre las direcciones IP genéricas, consulte Direcciones IP genéricas .

Referencia relacionada
SET CHLAUTH

Correlación de un ID de usuario confirmado por el cliente con un ID de usuario MCAUSER
Puede utilizar un registro de autenticación de canal para cambiar el atributo MCAUSER de un canal de conexión con el servidor, según el ID de usuario original 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 confirmado por el cliente.
usuario es el ID de usuario que se utilizará en lugar del nombre de usuario del cliente.

Referencia relacionada
SET CHLAUTH

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
) 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 WebSphere MQ estándar para los valores de SSLPEER. Consulte Reglas de WebSphere MQ para los valores de SSLPEER.
usuario es el ID de usuario que se utilizará para todas las conexiones que utilicen el DN especificado.

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 que se muestra a continuación, 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 para un ID de usuario confirmado por el cliente
Puede utilizar un registro de autenticación de canal para impedir que un ID de usuario confirmado por el cliente 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 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 confirmado por el cliente.

Referencia relacionada
SET CHLAUTH

Bloqueo del acceso para un Nombre distinguido SSL
Puede utilizar un registro de autenticación de canal para impedir que un Nombre distinguido SSL 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')
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 WebSphere MQ estándar para los valores de SSLPEER. Consulte Reglas de WebSphere MQ para los valores de SSLPEER.

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.

Procedimiento

- Para los sistemas UNIX, Linux y Windows, emita el mandato siguiente:

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

- Para IBM i, emita el mandato siguiente:

```
GRTMQMAUT OBJ('QMgrName') OBJTYPE(*MQM)  USER('GroupName') AUT(*CONNECT)
```

- Para z/OS, emita los siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

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 194.
La aplicación establece el contenido	Consulte “Otorgar autorización para establecer contexto” en la página 195.
La aplicación pasa el contexto	Consulte “Otorgar autorización para pasar contexto” en la página 196.
La aplicación transfiere mensajes a una cola agrupada en clúster	Consulte “Autorización de transferencia de mensajes a colas de clústeres remotos” en la página 252.
La aplicación transfiere mensajes a una cola local	Consulte “Otorgar autorización para transferir mensajes a una cola local” en la página 197.
La aplicación transfiere mensajes a una cola modelo	Consulte “Otorgar autorización para transferir mensajes a una cola modelo” en la página 197.
La aplicación transfiere mensajes a una cola remota	Consulte “Otorgar autorización para transferir mensajes a una cola de clúster remota” en la página 198.

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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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
```

- 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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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
```

- 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')
```

- Para z/OS, emita los mandatos siguientes 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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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 utiliza 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 197. 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 en el 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 sistemas UNIX, Linuxy Windows , también puede utilizar el mandato SET AUTHREC.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
RDEFINE MQQUEUE  QMgrNameObjectProfile UACC(NONE)
PERMIT QMgrNameObjectProfile CLASS(MQADMIN)
ID(GroupName) ACCESS(UPDATE)
```

Tenga en cuenta que puede usar el nombre del gestor de colas remotas (o el grupo de compartición de colas) sólo para las colas de clúste remotas.

Los nombres de las variables tienen los significados siguientes:

QMgrName
Nombre del gestor de colas. En z/OS, este valor también puede ser el nombre de un grupo de compartimiento 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 16. 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 199.
La aplicación se suscribe a un tema	Consulte “Otorgar autorización para suscribirse a temas” en la página 200.

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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento
- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento
- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

  ```
  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 también puede ser el nombre de un grupo de compartimiento 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.

Procedimiento

- Para los sistemas UNIX, Linux y 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 siguientes mandatos:

```
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 también puede ser el nombre de un grupo de compartimiento 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.

Autorización para administrar IBM WebSphere MQ en sistemas UNIX, Linux, and Windows

Los administradores de IBM WebSphere MQ pueden utilizar todos los mandatos de IBM WebSphere 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 consideraciones a los sistemas Windows.

Los administradores de IBM WebSphere MQ tienen autorización para utilizar todos los mandatos de WebSphere MQ (incluidos los mandatos para otorgar autorizaciones de WebSphere MQ para otros usuarios)

Para ser administrador de IBM WebSphere MQ, debe ser miembro de un grupo especial llamado grupo *mqm* (o ser miembro del grupo Administradores en sistemas Windows). El grupo mqm se crea automáticamente cuando se instala WebSphere MQ; añada más usuarios al grupo para permitirles que lleven a cabo tareas de administración. Todos los miembros de este grupo tienen acceso a todos los recursos. Este acceso sólo se puede revocar eliminando un usuario del grupo mqm y emitiendo el mandato REFRESH SECURITY. Los administradores pueden utilizar los mandatos de control para administrar WebSphere 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 WebSphere 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 ha otorgado 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 pueden utilizar el mandato **runmqsc** para emitir mandatos de script de IBM WebSphere 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. WebSphere MQ Explorer emite mandatos PCF para realizar tareas de administración. No es necesario que los administradores tengan autorizaciones adicionales si desean

utilizar WebSphere MQ Explorer para administrar un gestor de colas en el sistema local. Cuando se utiliza IBM WebSphere MQ Explorer para administrar un gestor de colas en otro sistema, los administradores deben tener las autorizaciones necesarias para que los gestores de colas remotos procesen los mandatos PCF.

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 se ejecutan en gestores de colas, colas, procesos, listas de nombres y objetos de información de autenticación, consulte Autorización para trabajar con objetos WebSphere 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

Además, en sistemas Windows , la cuenta SYSTEM tiene acceso completo a los recursos de WebSphere 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 WebSphere MQ son propiedad del ID de usuario.

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 mqm de dominio en el controlador de dominio 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 WebSphere 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 la página 207.

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, debe utilizar un indicador de mandatos elevado para emitir mandatos de administración de MQ, como **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 tareas 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 74).
- Emitir llamadas MQI desde un programa de aplicación (a menos que desee utilizar los enlaces de vía rápida en la llamada MQCONNX).
- Utilizar 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.
- Utilizar el mandato `dspmqtrc` para visualizar la salida de rastreo con formato de WebSphere 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 Versión 5.3 ha aumentado este límite, pero WebSphere 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, WebSphere MQ lo sustituye por el valor UNKNOWN. No defina un ID de usuario con un valor de UNKNOWN.

Gestión del grupo mqm

Se otorgan privilegios administrativos completos a los usuarios del grupo mqm a través de WebSphere 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 WebSphere MQ.

Estas tareas se describen en el apartado:

- Creación y gestión de grupos en Windows
- Creación y gestión de grupos en HP-UX
- Creación y gestión de grupos en AIX
- Creación y gestión de grupos en Solaris
- Creación y gestión de grupos en Linux

Si el controlador de dominio se ejecuta en Windows 2000 o Windows 2003, es posible que el administrador del dominio tenga que establecer una cuenta especial para que la utilice WebSphere MQ. Esto se describe en la sección Configuración de cuentas de WebSphere MQ.

Autorización para trabajar con objetos IBM WebSphere MQ en sistemas UNIX, Linux, and Windows

Todos los objetos están protegidos por IBM WebSphere 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 WebSphere 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 WebSphere MQ en sistemas UNIX, Linux, and Windows” en la página 202

Los administradores de IBM WebSphere MQ pueden utilizar todos los mandatos de IBM WebSphere 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 consideraciones a los sistemas Windows.

Cuando se realizan comprobaciones de seguridad en sistemas 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, WebSphere 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 WebSphere MQ; WebSphere 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 WebSphere 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).

WebSphere MQ comprueba el recurso especificado 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 94.

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 debidamente autorizada puede emitir una llamada **MQOPEN** especificando un ID de usuario alternativo; a continuación, se realizan comprobaciones de control de acceso 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 subscribe a un tema, se realizan las comprobaciones de autorización en relación con los objetos de tema que se encuentran en el árbol de temas en, o por encima de, el punto del árbol de temas en el 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.

Cómo IBM WebSphere MQ implementa el control de accesos en sistemas UNIX, Linux, and Windows

IBM WebSphere MQ utiliza los servicios de seguridad proporcionados por el sistema operativo subyacente mediante el gestor de autorizaciones sobre objetos. IBM WebSphere 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 WebSphere MQ. WebSphere 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 172). 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 WebSphere MQ si tienen la autorización correcta. En el apartado "Control del acceso a los objetos mediante el OAM en sistemas UNIX, Linux y Windows" en la página 164 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 inician sesión con el ID del sistema. El acceso de usuarios a la cola no se puede cambiar.

WebSphere MQ suministra mandatos para crear y mantener listas de control de accesos. Para obtener más información acerca de estos mandatos, consulte "Control del acceso a los objetos mediante el OAM en sistemas UNIX, Linux y Windows" en la página 164.

WebSphere MQ pasa al OAM una petición 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. WebSphere MQ sigue la decisión adoptada por el OAM; si el OAM no puede tomar una decisión, WebSphere MQ no permite el acceso.

Identificación del ID de usuario en sistemas 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 WebSphere 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 WebSphere 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 sistemas UNIX, Linux y Windows" en la página 209. Las aplicaciones no pueden alterar esta información a menos que tengan autorización para cambiar la información de contexto.

Principales y grupos

Los principales pueden pertenecer a grupos. Puede otorgar acceso a un recurso determinado a grupos en vez de a usuarios individuales, a fin de reducir la cantidad de administración necesaria. En los sistemas UNIX and Linux , todas las listas de control de acceso (ACL) se basan en grupos, pero en los sistemas Windows , las ACL se basan en ID de usuario y grupos.

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 se describe en:

- Creación y gestión de grupos en Windows
- Creación y gestión de grupos en HP-UX
- Creación y gestión de grupos en AIX
- Creación y gestión de grupos en Solaris
- 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 modo que todos los cambios que realice en los miembros de grupo del principal no se reconocen hasta que no se reinicia el gestor de colas, a menos que emita el mandato MQSC REFRESH SECURITY (o el mandato PCF equivalente).

Sistemas UNIX and Linux

Todas las ACL están basadas en grupos. Cuando a un usuario se le otorga acceso a un recurso determinado, en la ACL se incluye el grupo primario del ID de usuario. 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 WebSphere 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 un ID de usuario es demasiado largo, por lo que los ID de usuario arbitrarios utilizarán las autorizaciones de acceso de UNKNOWN.

Los ID de usuario tener 12 caracteres como máximo y los nombres de grupo también.

Sistemas Windows

Las ACL están basadas en los grupos y en los ID de usuario. Las comprobaciones son las mismas que para los sistemas UNIX con la excepción de que los ID de usuario individuales también pueden mostrarse en la ACL. Puede tener usuarios diferentes en dominios diferentes con el mismo ID de usuario. WebSphere MQ permite que los ID de usuario se califiquen mediante un nombre de dominio para que a estos usuarios se les puedan otorgar distintos niveles de acceso.

El nombre del grupo puede incluir opcionalmente un nombre de dominio, especificado con los formatos siguientes:

```
GroupName@domain
domain\GroupName
```

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 WebSphere 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.

Identificadores de seguridad (SID) de Windows

WebSphere MQ en Windows utiliza el SID donde está disponible. Si un SID de Windows no se suministra con una petición de autorización, WebSphere MQ identifica el usuario basándose simplemente en el nombre de usuario solamente, pero esto podría ocasionar que se otorgara una 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 WebSphere MQ para Windows, WebSphere MQ almacena el SID en el descriptor de mensaje. Cuando WebSphere 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 petición de autorización, WebSphere MQ identifica al usuario basándose simplemente 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 de WebSphere MQ incorrecta. Para evitar este problema, incluya un SID en cada petición de autorización; WebSphere 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.

Autorización de usuario alternativo en sistemas UNIX, Linux y Windows

Puede especificar que un ID de usuario pueda utilizar la autorización de otro usuario cuando accede a un objeto WebSphere MQ. Esto se denomina *autorización de usuario alternativo* y puede utilizarla en cualquier objeto WebSphere 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.

Autorización de contexto en sistemas UNIX, Linux y 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 cualquiera de las opciones de contexto en cualquier llamada 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 WebSphere 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. *MCAUserIdentifier* contiene los 12 primeros bytes del identificador de usuario MCA. Si el identificador de usuario MCA no está en blanco, especifica el identificador de usuario que utilizará el agente de canal de mensajes para la autorización para acceder a recursos de MQ. Asegúrese de que MCAUSER tenga menos de 12 caracteres en la plataforma Windows.

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.

En los sistemas IBM i, UNIX, Linux y Windows , 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.

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 WebSphere 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* . Asimismo, en un canal MQI, una salida de seguridad situada en el extremo del cliente del canal puede enviar el ID de usuario asociado con la aplicación cliente de WebSphere MQ. 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 SSL 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 WebSphere MQ v7.1 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 WebSphere MQ

En las conexiones de cliente MQI de WebSphere MQ, 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 WebSphere 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 153.

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 WebSphere 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.

Confidencialidad de mensajes

Para mantener la confidencialidad, cifre los mensajes. Existen diversos métodos de cifrado de mensajes en WebSphere 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 WebSphere MQ 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 SSL o TLS, o puede escribir su propia salida de seguridad, salida de mensaje o programas de salida de emisión y recepción.

Para obtener más información sobre WebSphere MQ Advanced Message Security, consulte "Planificación de Advanced Message Security" en la página 66. El uso de SSL y TLS con WebSphere MQ se describe en "Soporte de IBM WebSphere MQ para SSL y TLS" 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 231.

Conexión de dos gestores de colas utilizando SSL o TLS

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 obtener y gestionar los certificados digitales. En un sistema de prueba, puede utilizar certificados o certificados autofirmados emitidos por una entidad emisora de certificados (CA) local. En un sistema de producción, no utilice certificados autofirmados. Para obtener más información, consulte ..//zs14140_.dita.

Para obtener información detallada sobre la creación y la gestión de certificados, consulte "Trabajar con SSL o TLS en sistemas UNIX, Linux, and Windows" en la página 118.

Esta colección de temas presentan las tareas que forman parte de la configuración de las comunicaciones SSL y se proporciona una guía paso a paso sobre cómo 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 WebSphere MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

Notas:

1. En este contexto, un cliente SSL hace referencia a la conexión iniciando el reconocimiento.
2. Consulte el Glosario para obtener más detalles.

En sistemas UNIX, Linux y Windows, el cliente SSL o TLS envía un certificado sólo si tiene uno etiquetado en el formato WebSphere MQ correcto, que es `ibmwebspheremq` seguido por el nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`.

WebSphere 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 no se realiza correctamente sólo si el extremo del canal que actúa como el servidor SSL o TLS se ha definido con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información sobre la conexión de un gestor de colas de forma anónima, es decir, cuando el cliente SSL o TLS no envía un certificado, consulte "Conexión de dos gestores de colas utilizando autenticación unidireccional" en la página 216.

Utilización de certificados autofirmados para la autenticación mutua de dos gestores de colas

Siga estas instrucciones de ejemplo para implementar la autenticación mutua entre dos gestores de colas, utilizando certificados SSL o TLS autofirmados.

Acerca de esta tarea

Escenario:

- Sean dos gestores de colas, QM1 y QM2, que deben comunicarse de forma segura. Se necesita una autenticación mutua entre QM1 y QM2.
- Se ha decidido probar la comunicación segura utilizando certificados autofirmados.

La configuración resultante es parecida a la siguiente:

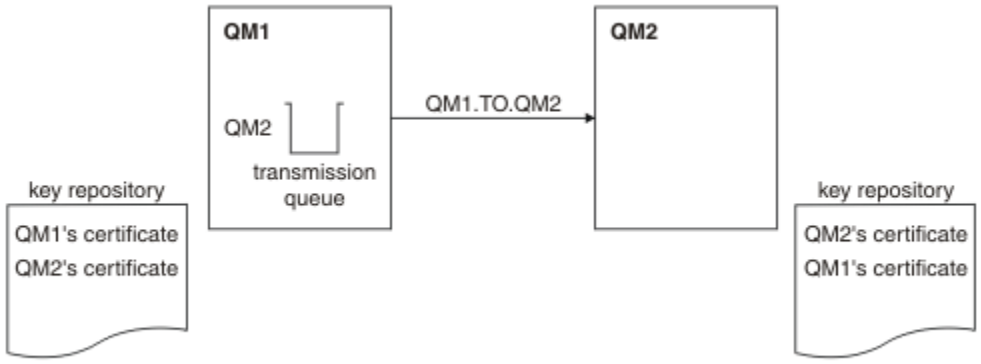


Figura 14. Configuración resultante de esta tarea

En Figura 14 en la página 213, el repositorio de claves para QM1 contiene el certificado para QM1 y el certificado público de QM2. El repositorio de claves para QM2 contiene el certificado para QM2 y el certificado público de QM1.

Procedimiento

1. Prepare el repositorio de claves en cada gestor de colas según el sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Cree un certificado autofirmado para cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
3. Extraiga una copia de cada certificado:
   - En sistemas UNIX, Linux y Windows.
4. Transfiera la parte pública del certificado de QM1 al sistema de QM2 y viceversa, utilizando un programa de utilidad como FTP.
5. Agregue el certificado de asociado al repositorio de claves por cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
6. En QM1, defina un canal emisor y la cola de transmisión asociada en el gestor de colas emitiendo mandatos como en el siguiente ejemplo:

   ```
   DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
   SSLCIPH(RC4_MD5_US) DESCR('Sender channel using SSL from QM1 to QM2')

   DEFINE QLOCAL(QM2) USAGE(XMITQ)
   ```

   Este ejemplo utiliza la CipherSpec RC4_MD5. Las CipherSpecs de cada extremo del canal deben ser la misma.
7. En QM2, defina un canal receptor emitiendo un mandato como en el siguiente ejemplo:

   ```
   DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from QM1 to QM2')
   ```

   El canal debe tener el mismo nombre que el canal emisor que ha definido en el paso 6 y utilizar la misma CipherSpec.
8. Inicie el canal.

Resultados

Se crean los depósitos de claves y los canales, tal como se ilustra en la Figura 14 en la página 213

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante será similar a la mostrada en los ejemplos siguientes.

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM1.TO.QM2)                     CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=QM2,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

Desde el gestor de colas QM2, entre el siguiente mandato:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM2.TO.QM1)                     CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QM1)
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

En cada caso, el valor de SSLPEER debe coincidir con el del DN en el certificado de socio que se ha creado en el paso 2. El nombre del emisor coincide con el nombre de igual porque el certificado es autofirmado.

SSLPEER es opcional. Si se especifica, su valor debe establecerse para que se permita el DN del certificado de asociado (creado en el paso 2). Para obtener más información sobre el uso de SSLPEER, consulte WebSphere MQ rules for SSLPEER values .

Utilización de certificados autofirmados por CA para la autenticación mutua de dos gestores de colas

Siga estas instrucciones de ejemplo para implementar la autenticación mutua entre dos gestores de colas, utilizando certificados SSL o TLS firmados por una CA.

Acerca de esta tarea

Escenario:

- Dispone de dos gestores de colas denominados QMA y QMB, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre QMA y QMB.
- En el futuro está previsto utilizar esta red en un entorno de producción y, por consiguiente, se ha decidido utilizar certificados firmados por CA desde el principio.

La configuración resultante es parecida a la siguiente:

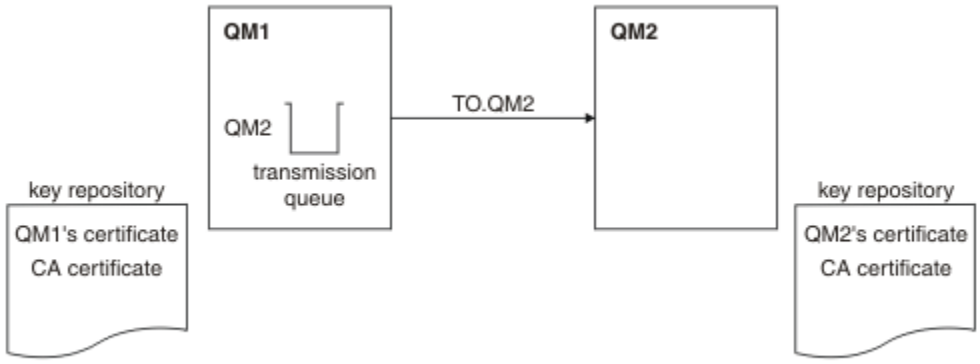


Figura 15. Configuración resultante de esta tarea

En la Figura 15 en la página 215, el repositorio de claves para QMA contiene el certificado de QMA y el certificado CA. El repositorio de claves para QMB contiene el certificado de QMB y el certificado CA. En este ejemplo, tanto el certificado de QMA como el certificado de QMB se emitieron mediante la misma CA (entidad emisora de certificados). Si el certificado de QMA y el certificado de QMB los emitieron CA diferentes, los repositorios de claves para QMA y QMB contendrán ambos certitificados de CA.

Procedimiento

1. Prepare el repositorio de claves en cada gestor de colas según el sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Solicite un certificado firmado por una CA para cada gestor de colas.

   Puede utilizar CA diferentes para los dos gestores de colas.
   - En sistemas UNIX, Linux y Windows.
3. Añada el certificado de una entidad emisora de certificados al repositorio de claves para cada gestor de colas:

   Si los gestores de colas utilizan entidades emisoras de certificados (CA) diferentes, el certificado de CA de cada entidad emisora de certificados deberá añadirse a ambos repositorios de claves.
   - En sistemas UNIX, Linux y Windows.
4. Añada el certificado firmado por la CA en el repositorio de claves para cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
5. En QMA, defina un canal emisor y la cola de transmisión asociada en el gestor de colas emitiendo mandatos como en el siguiente ejemplo:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
   SSLCIPH(RC2_MD5_EXPORT) DESCR('Sender channel using SSL from QMA to QMB')

   DEFINE QLOCAL(QMB) USAGE(XMITQ)
   ```

   Este ejemplo utiliza la CipherSpec RC4_MD5. Las CipherSpecs de cada extremo del canal deben ser la misma.
6. En QMB, defina un canal receptor emitiendo un mandato como en el siguiente ejemplo:

   ```
   DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
   ```

   El canal debe tener el mismo nombre que el canal emisor que ha definido en el paso 6 y utilizar la misma CipherSpec.
7. Inicie el canal:

Resultados

Se crean repositorios de claves y canales, como se ilustra en Figura 15 en la página 215.

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante es parecida a la que se muestra en los ejemplos siguientes.

Desde el gestor de colas QMA, entre el siguiente mandato:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QMB)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QMB)
```

Desde el gestor de colas QMB, entre el siguiente mandato:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

En cada caso, el valor de SSLPEER debe coincidir con el del nombre distinguido (DN) en el certificado de socio que se ha creado en el paso 2. El nombre del emisor coincide con el DN de sujeto del certificado de CA que ha firmado el certificado personal añadido en el Paso 4.

Conexión de dos gestores de colas utilizando autenticación unidireccional

Siga estas instrucciones de ejemplo para modificar un sistema con la autenticación mutua para permitir a un gestor de colas conectarse con otro utilizando la autenticación unidireccional; es decir, cuando el cliente SSL o TLS no envía un certificado.

Acerca de esta tarea

Escenario:

- Los dos gestores de colas (QM1 y QM2) se han configurado como en “Utilización de certificados autofirmados por CA para la autenticación mutua de dos gestores de colas” en la página 214.
- Desea cambiar QM1 para que se conecte a QM2 utilizando la autenticación unidireccional.

La configuración resultante es parecida a la siguiente:

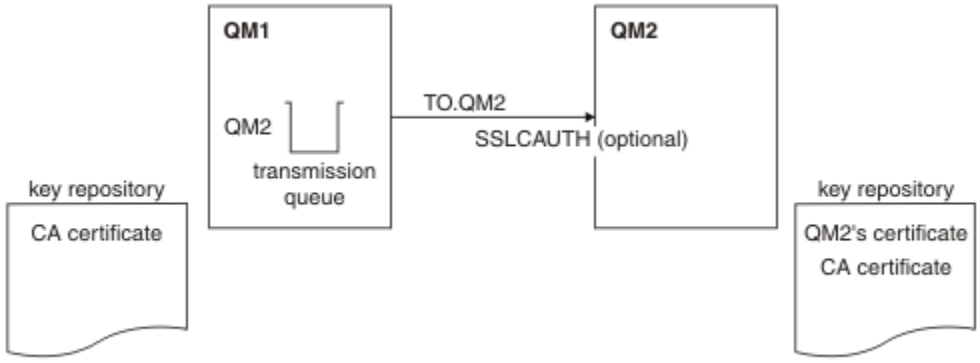


Figura 16. Gestores de colas que permiten la autenticación unidireccional

Procedimiento

1. Elimine el certificado personal de QM1 de su repositorio de claves, de acuerdo con el sistema operativo:
   - En los sistemas UNIX, Linux y Windows. El certificado se etiqueta como se indica a continuación:
     - `ibmwebspheremq` seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`.
2. Opcional: En QM1, si algún canal SSL o TLS se ha ejecutado anteriormente, renueve el entorno SSL o TLS.
3. Permitir conexiones anónimas en el receptor.

Resultados

Los repositorios de claves y los canales cambian como se ilustra en Figura 16 en la página 217.

Qué hacer a continuación

Si el canal emisor estaba en ejecución y ha emitido el mandato REFRESH SECURITY TYPE(SSL) (en el paso 2), el canal se reiniciará automáticamente. Si el canal emisor no estaba en ejecución, inícielo.

En el extremo del canal del servidor, la presencia del valor de parámetro de nombre de igual en la visualización del estado del canal indica que se ha emitido un certificado de cliente.

Compruebe que la tarea se ha completado satisfactoriamente emitiendo algunos mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante es similar a la que se muestra en los ejemplos siguientes:

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
```

```
   XMITQ(QM2)
```

Desde el gestor de colas QM2 ejecute el siguiente mandato:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)                            SSLCERTI( )
   SSLPEER( )                              STATUS(RUNNING)
   SUBSTATE(RECEIVE)                       XMITQ( )
```

En QM2, el campo SSLPEER está vacío, lo que muestra que QM1 no ha enviado un certificado. En QM1, el valor de SSLPEER coincide con el del DN del certificado personal de QM2.

Conexión de un cliente a un gestor de colas de forma segura

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 obtener y gestionar los certificados digitales. En un sistema de prueba, puede utilizar certificados o certificados autofirmados emitidos por una entidad emisora de certificados (CA) local. En un sistema de producción, no utilice certificados autofirmados. Para obtener más información, consulte ..//zs14140_.dita.

Para obtener información detallada sobre la creación y la gestión de certificados, consulte “Trabajar con SSL o TLS en sistemas UNIX, Linux, and Windows” en la página 118.

Esta colección de temas presentan las tareas que forman parte de la configuración de las comunicaciones SSL y se proporciona una guía paso a paso sobre cómo 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 WebSphere MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En sistemas UNIX, Linux, and Windows, el cliente SSL o TLS envía un certificado sólo si tiene uno etiquetado en el formato WebSphere MQ correcto, que es `ibmwebspheremq` seguido por el ID de usuario de inicio de sesión en minúsculas, por ejemplo `ibmwebspheremqmyuserid`.

WebSphere 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 no se realiza correctamente sólo si el extremo del canal que actúa como el servidor SSL o TLS se ha definido con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información sobre la conexión de un gestor de colas de forma anónima, consulte “Conexión de un cliente a un gestor de colas de forma anónima” en la página 222.

Utilización de certificados autofirmados para la autenticación mutua de un cliente y un gestor de colas

Siga las instrucciones de este ejemplo para implementar la autenticación mutua entre un cliente y un gestor de colas, utilizando certificados SSL o TLS autofirmados.

Acerca de esta tarea

Escenario:

- Tiene un cliente, C1, y un gestor de colas, QM1, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre C1 y QM1.
- Ha decidido probar la comunicación segura utilizando certificados autofirmados.

DCM en IBM i no admite certificados autofirmados, por lo que esta tarea no se aplica en sistemas IBM i.

La configuración resultante es parecida a la siguiente:

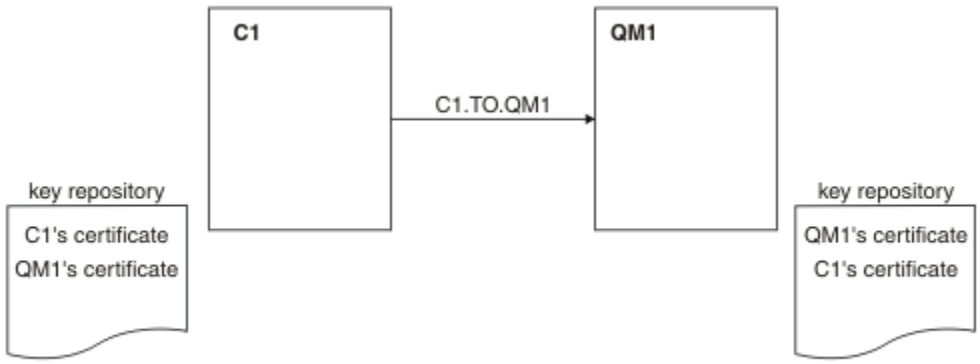


Figura 17. Configuración resultante de esta tarea

En Figura 17 en la página 219, el repositorio de claves para QM1 contiene el certificado para QM1 y el certificado público para C1. El repositorio de claves para C1 contiene el certificado para C1 y el certificado público de QM1.

Procedimiento

1. Prepare el repositorio de claves en el cliente y el gestor de colas conforme al sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Cree certificados autofirmados para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
3. Extraiga una copia de cada certificado:
   - En sistemas UNIX, Linux y Windows.
4. Transfiera la parte pública del certificado de C1 al sistema de QM1 y viceversa, utilizando un programa de utilidad como FTP.
5. Añada el certificado de socio al repositorio de claves para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
6. Emita el mandato REFRESH SECURITY TYPE (SSL) en el gestor de colas.
7. Defina un canal de conexión con el cliente siguiendo uno de estos métodos:
   - Utilizando la llamada MQCONNX con la estructura MQSCO en C1, tal como se describe en Creación de un canal de conexión-cliente en el cliente MQI de WebSphere MQ.

- Utilizando una tabla de definiciones de canal de cliente, tal como se describe en Creación de definiciones de conexión de servidor y conexión de cliente en el servidor .

8. En QM1, defina un canal de conexión con el servidor, utilizando un mandato como el del ejemplo siguiente:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
```

   El canal debe tener el mismo nombre que el canal de conexión de cliente que ha definido en el paso 6, y utilizar la misma especificación de cifrado (CipherSpec).

Resultados

Se crean los repositorios de claves y los canales, tal como se ilustra en la Figura 17 en la página 219

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida del resultado es parecida a la que se muestra en el siguiente ejemplo.

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                   CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                    CURRENT
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                        SUBSTATE(RECEIVE)
```

Es opcional establecer el atributo de filtro SSLPEER de las definiciones de canal. Si se establece la definición de canal SSLPEER, su valor debe coincidir con el DN de sujeto en el certificado de socio que se ha creado en el paso 2. Después de una conexión satisfactoria, el campo SSLPEER de la salida DISPLAY CHSTATUS muestra el DN de asunto del certificado de cliente remoto.

Utilización de certificados firmados por CA para la autenticación mutua de un cliente y un gestor de colas

Siga las instrucciones de este ejemplo para implementar la autenticación mutua entre un cliente y un gestor de colas, utilizando certificados SSL o TLS firmados por CA.

Acerca de esta tarea

Escenario:

- Tiene un cliente, C1, y un gestor de colas, QM1, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre C1 y QM1.
- En el futuro está previsto utilizar esta red en un entorno de producción y, por consiguiente, se ha decidido utilizar certificados firmados por CA desde el principio.

La configuración resultante es parecida a la siguiente:

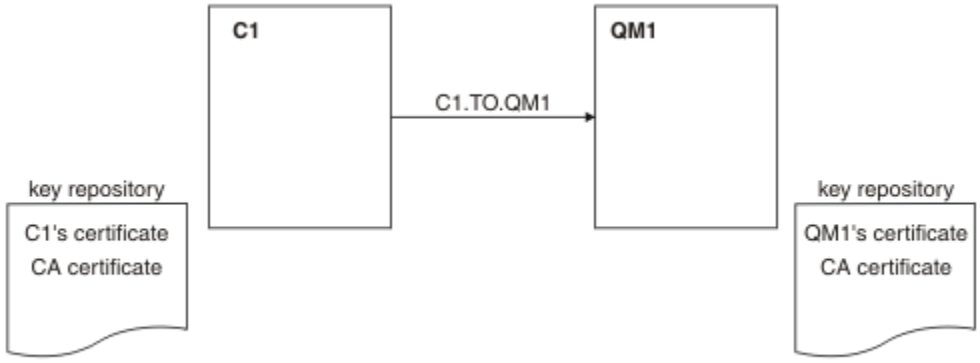


Figura 18. Configuración resultante de esta tarea

En Figura 18 en la página 221, el repositorio de claves para C1 contiene el certificado para C1 y el certificado de CA. El repositorio de claves de QM1 contiene el certificado de QM1 y el certificado de la CA. En este ejemplo, tanto el certificado de C1 como el certificado de QM1 se emitieron mediante la misma CA (entidad emisora de certificados). Si el certificado de C1 y el certificado de QM1 los emitieron CA diferentes, los repositorios de claves para C1 y QM1 contendrán ambos certificados de CA.

Procedimiento

1. Prepare el repositorio de claves en el cliente y el gestor de colas conforme al sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Solicite un certificado firmado por CA para el cliente y el gestor de colas.

   Puede utilizar autoridades emisoras de certificados (CA) distintas para el cliente y el gestor de colas.
   - En sistemas UNIX, Linux y Windows.
3. Añada el certificado de la entidad emisora de certificados al repositorio de claves para el cliente y el gestor de colas.

   Si el cliente y los gestores de colas utilizan entidades emisoras de certificados (CA) diferentes, el certificado de CA de cada entidad emisora de certificados debe añadirse a ambos repositorios de claves.
   - En sistemas UNIX, Linux y Windows.
4. Añada el certificado firmado por CA al repositorio de claves para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
5. Defina un canal de conexión con el cliente siguiendo uno de estos métodos:
   - Utilizando la llamada MQCONNX con la estructura MQSCO en C1, tal como se describe en Creación de un canal de conexión-cliente en el cliente MQI de WebSphere MQ.
   - Utilizando una tabla de definiciones de canal de cliente, tal como se describe en Creación de definiciones de conexión de servidor y conexión de cliente en el servidor .
6. En QM1, defina un canal de conexión con el servidor emitiendo un mandato como el del ejemplo siguiente:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   El canal debe tener el mismo nombre que el canal de conexión de cliente que ha definido en el paso 6, y utilizar la misma especificación de cifrado (CipherSpec).

Resultados

Se crean repositorios de claves y canales, como se ilustra en Figura 18 en la página 221.

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida resultante es parecida a la que se muestra en el siguiente ejemplo.

Desde el gestor de colas QM1, especifique el siguiente mandato:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

El campo SSLPEER de la salida DISPLAY CHSTATUS muestra el DN de sujeto del certificado de cliente remoto que se ha creado en el Paso 2. El nombre del emisor coincide con el DN de sujeto del certificado de CA que ha firmado el certificado personal añadido en el Paso 4.

Conexión de un cliente a un gestor de colas de forma anónima

Siga las instrucciones de este ejemplo para modificar un sistema con autenticación mutua a fin de permitir que un gestor de colas se conecte a otro de forma anónima.

Acerca de esta tarea

Escenario:

- Su gestor de colas y cliente (QM1 y C1) se han configurado como se indica en “Utilización de certificados firmados por CA para la autenticación mutua de un cliente y un gestor de colas” en la página 220.
- Desea cambiar C1 de modo que se conecte de forma anónima a QM1.

La configuración resultante es parecida a la siguiente:

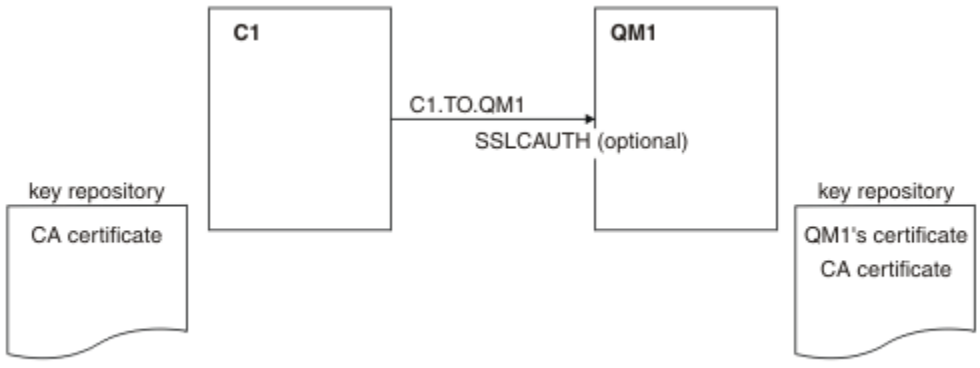


Figura 19. Cliente y gestor de colas que permiten conexión anónima

Procedimiento

1. Elimine el certificado personal del repositorio de claves para C1, conforme al sistema operativo:
   - En los sistemas UNIX, Linux y Windows. El certificado se etiqueta como se indica a continuación:

- ibmwebspheremq seguido por el ID de usuario de inicio de sesión doblado a minúsculas, por ejemplo ibmwebspheremqmyuserid.

2. Reinicie la aplicación cliente o haga que la aplicación cliente se cierre y abra de nuevo todas las conexiones SSL o TLS.
3. Permita las conexiones anónimas en el gestor de colas emitiendo el siguiente mandato:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Resultados

Los repositorios de claves y los canales cambian como se ilustra en Figura 19 en la página 222.

Qué hacer a continuación

En el extremo del canal del servidor, la presencia del valor de parámetro de nombre de igual en la visualización del estado del canal indica que se ha emitido un certificado de cliente.

Compruebe que la tarea se ha completado satisfactoriamente emitiendo algunos mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida del resultado es parecida a la que se muestra en el siguiente ejemplo:

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI( )                             SSLPEER( )
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

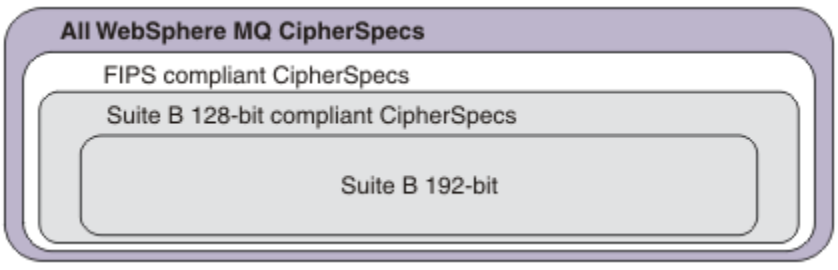
Los campos SSLCERTI y SSLPEER están vacíos, lo que indica que C1 no ha enviado ningún certificado.

Especificación de CipherSpecs

Especifique una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC de **DEFINE CHANNEL** o en el mandato MQSC de **ALTER CHANNEL** .

Algunas de las CipherSpecs que puede utilizar con IBM WebSphere MQ son compatibles con FIPS. Otras, como por ejemplo, NULL_MD5 no lo son. Asimismo, algunas de las CipherSpecs compatibles con FIPS también son compatibles con Suite B, aunque otras 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:



Las especificaciones de cifrado que puede utilizar con el soporte de SSL y TLS de IBM WebSphere MQ aparecen listadas en la tabla siguiente. 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 SSL es el tamaño almacenado en el certificado, a menos que esté determinado por la CipherSpec, tal como está indicado en la tabla.

Nombre de CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de cifrado	Bits de cifrado	FIPS[1]	Suite B de 128 bits	Suite B de 192 bits
NULL_MD5 [a]	SSL 3.0	MD5	Ninguna	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Ninguna	0	No	No	No
RC4_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC4	40	No	No	No
RC4_MD5_US [a]	SSL 3.0	MD5	RC4	128	No	No	No
RC4_SHA_US [a]	SSL 3.0	SHA-1	RC4	128	No	No	No
RC2_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC2	40	No	No	No
DES_SHA_EXPORT [2] [a]	SSL 3.0	SHA-1	DES	56	No	No	No
RC4_56_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	RC4	56	No	No	No
DES_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	DES	56	No	No	No
TLS_RSA_WITH_AES_128_CBC_SHA [a]	TLS 1.0	SHA-1	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sí	No	No
TLS_RSA_WITH_DES_CBC_SHA [a]	TLS 1.0	SHA-1	DES	56	No[5]	No	No
FIPS_WITH_DES_CBC_SHA [b]	SSL 3.0	SHA-1	DES	56	No[6]	No	No
TLS_RSA_WITH_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sí	No	No
ECDHE_ECDSA_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No
ECDHE_RSA_RC4_128_SHA256 [b]	TLS 1.2	SHA_1	RC4	128	No	No	No
ECDHE_ECDSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sí	No	No
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sí	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	Sí	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	Sí

Nombre de CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de cifrado	Bits de cifrado	FIPS[1]	Suite B de 128 bits	Suite B de 192 bits
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Ninguna	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Ninguna	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Ninguna	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Ninguna	Ninguna	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Notas:

1. 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.
2. 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.
3. El tamaño de clave de reconocimiento es de 1024 bits.
4. Este CipherSpec no se puede utilizar para garantizar una conexión desde WebSphere MQ Explorer a un gestor de colas amenos que se apliquen los archivos de políticas no restringidas apropiados al JRE utilizado por Explorer.
5. Esta CipherSpec obtuvo el certificado FIPS 140-2 antes del 19 mayo de 2007.
6. 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.
7. Se puede utilizar esta CipherSpec para transferir hasta 32 GB de datos antes de que la conexión concluya con el error AMQ9288. Para evitar este error, evite utilizar triple DES, o habilite el restablecimiento de clave secreta cuando utilice esta CipherSpec.

Soporte de plataformas:

- a Disponible en todas las plataformas soportadas.
- b Disponible sólo en plataformas UNIX, Linux, and Windows.

Conceptos relacionados

“Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ” en la página 36
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 WebSphere MQ.

Referencia relacionada

DEFINE CHANNEL
ALTER CHANNEL

CipherSpecs en desuso

Una lista de CipherSpecs en desuso que puede utilizar con WebSphere MQ, si es necesario.

Consulte “Valores de CipherSpec soportados en IBM WebSphere MQ” en la página 40 para obtener más información sobre cómo habilitar las CipherSpecs en desuso.

Las CipherSpecs en desuso que se pueden utilizar con el soporte TLSde WebSphere MQ se listan en la siguiente tabla:

Soporte de plataforma “1” en la página 228	Nombre de CipherSpec	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado	Bits de cifrado	FIPS “2” en la página 228	Suite B	Actualizar cuando esté en desuso
Todo	DES_SHA_EXPORT “3” en la página 228	SSL 3.0	SHA-1	DES	56	No	No	7.5.0.6
Windows UNIX Linux	DES_SHA_EXPORT1024 “4” en la página 228	SSL 3.0	SHA-1	DES	56	No	No	7.5.0.6
Windows UNIX Linux	FIPS_WITH_DES_CBC_SHA	SSL 3.0	SHA-1	DES	56	No “6” en la página 228	No	7.5.0.6
Windows UNIX Linux	FIPS_WITH_3DES_EDE_CBC_SHA	SSL 3.0	SHA-1	3DES	168	No “7” en la página 228	No	7.5.0.8
Todo	NULL_MD5	SSL 3.0	MD5	Ninguna	0	No	No	7.5.0.6
Todo	NULL_SHA	SSL 3.0	SHA-1	Ninguna	0	No	No	7.5.0.6
Todo	RC2_MD5_EXPORT “3” en la página 228	SSL 3.0	MD5	RC2	40	No	No	7.5.0.7
Todo	RC4_MD5_EXPORT “3” en la página 228	SSL 3.0	MD5	RC4	40	No	No	7.5.0.7
Todo	RC4_MD5_US	SSL 3.0	MD5	RC4	128	No	No	7.5.0.7
Todo	RC4_SHA_US	SSL 3.0	SHA-1	RC4	128	No	No	7.5.0.7
Windows UNIX Linux	RC4_56_SHA_EXPORT1024 “4” en la página 228	SSL 3.0	SHA-1	RC4	56	No	No	7.5.0.7
Todo	TRIPLE_DES_SHA_US	SSL 3.0	SHA-1	3DES	168	No	No	7.5.0.8
Todo	TLS_RSA_WITH_DES_CBC_SHA	TLS 1.0	SHA-1	DES	56	No “5” en la página 228	No	7.5.0.6
Windows UNIX Linux	ECDHE_ECDSA_NULL_SHA256	TLS 1.2	SHA-1	Ninguna	0	No	No	7.5.0.6

Soporte de plataforma "1" en la página 228	Nombre de CipherSpec	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado	Bits de cifrado	FIPS "2" en la página 228	Suite B	Actualizar cuando esté en desuso
Windows UNIX Linux	ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
Windows UNIX Linux	ECDHE_RSA_NULL_SHA256	TLS 1.2	SHA-1	Ninguna	0	No	No	7.5.0.6
Windows UNIX Linux	ECDHE_RSA_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
Windows UNIX Linux	TLS_RSA_WITH_NULL_NULL	TLS 1.2	Ninguna	Ninguna	0	No	No	7.5.0.6
Todo	TLS_RSA_WITH_NULL_SHA256	TLS 1.2	SHA-256	Ninguna	0	No	No	7.5.0.6
Windows UNIX Linux	TLS_RSA_WITH_RC4_128_SHA256	TLS 1.2	SHA-1	RC4	128	No	No	7.5.0.7
Todo	TLS_RSA_WITH_3DES_EDE_CBC_SHA"8" en la página 228	TLS 1.0	SHA-1	3DES	168	Sí	No	7.5.0.8
Windows UNIX Linux	ECDHE_ECDSA_3DES_EDE_CBC_SHA256"8" en la página 228	TLS 1.2	SHA-1	3DES	168	Sí	No	7.5.0.8
Windows UNIX Linux	ECDHE_RSA_3DES_EDE_CBC_SHA256"8" en la página 228	TLS 1.2	SHA-1	3DES	168	Sí	No	7.5.0.8

Soporte de plataforma "1" en la página 228	Nombre de CipherSpec	Protocolo utilizado	Integridad de datos	Algoritmo de cifrado	Bits de cifrado	FIPS "2" en la página 228	Suite B	Actualizar cuando esté en desuso

Notas:

1. Si no se indica ninguna plataforma específica, CipherSpec está disponible en todas las plataformas.
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. 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.
4. El tamaño de clave de reconocimiento es de 1024 bits.
5. Esta CipherSpec obtuvo el certificado FIPS 140-2 antes del 19 mayo de 2007.
6. 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 esta CipherSpec anteriormente cumplía (ahora ya no) la normativa FIPS. Esta CipherSpec está en desuso y su uso no se recomienda.
7. El nombre `FIPS_WITH_3DES_EDE_CBC_SHA` es histórico y refleja el hecho de que esta CipherSpec anteriormente cumplía (ahora ya no) la normativa FIPS. Esta CipherSpec está en desuso.
8. Se puede utilizar esta CipherSpec para transferir hasta 32 GB de datos antes de que la conexión concluya con el error AMQ9288. Para evitar este error, evite utilizar triple DES o habilite el restablecimiento de clave secreta cuando utilice esta CipherSpec.

Obtención de información sobre CipherSpecs utilizando IBM WebSphere MQ Explorer

Puede utilizar IBM WebSphere MQ Explorer para visualizar descripciones de CipherSpecs.

Utilice el procedimiento siguiente para obtener información acerca de las CipherSpecs que aparecen en la "Especificación de CipherSpecs" en la página 223:

1. Abra **IBM WebSphere 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 SSL, 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 WebSphere 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 SSL, es posible que el sistema dé soporte a nuevas CipherSpecs que no figuran en la "Especificación de CipherSpecs" en la página

223. 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 CipherSpec de SSL que sea válida y que esté soportada por la versión de SSL en la que se esté ejecutando el sistema.

IBM i
Una serie de dos caracteres que representa un valor hexadecimal.

Para obtener más información sobre los valores permitidos, consulte la documentación del producto adecuada (busque *cipher_spec* en la documentación del productoIBM i).

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 dos caracteres que representa un valor hexadecimal. Los códigos hexadecimales se corresponden con los valores definidos en el protocolo SSL.

Para obtener más información, consulte la descripción de gsk_environment_open() en el capítulo de consultas de API de la publicación *z/OS Cryptographic Services System SSL Programming*, SC24-5901, donde hay una lista de todas las especificaciones de cifrado SSL V3.0 y TLS V1.0 soportadas en formato de códigos hexadecimales de dos dígitos.

Consideraciones para los clústeres de WebSphere MQ

Con los clústeres de WebSphere MQ, es más seguro utilizar los nombres de CipherSpec de la "Especificación de CipherSpecs" en la página 223. Si utiliza una especificación alternativa, tenga en cuenta que la especificación puede no ser válida en otras plataformas. Para obtener más información, consulte "SSL y clústeres" en la página 254.

Especificación de una CipherSpec para un cliente MQI de IBM WebSphere MQ

Tiene tres opciones para especificar una CipherSpec para un cliente MQI de IBM WebSphere MQ .

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 clases IBM WebSphere MQ para Java y clases IBM WebSphere MQ para JMS

Las clases IBM WebSphere MQ para Java y las clases IBM WebSphere MQ para JMS especifican CipherSuites de forma diferente a otras plataformas.

Para obtener información sobre cómo especificar una CipherSuite con clases IBM WebSphere MQ para Java, consulte Soporte SSL (Secure Sockets Layer).

Para obtener información sobre cómo especificar una CipherSuite con clases IBM WebSphere MQ para JMS, consulte Utilización de SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS.

Restablecimiento de claves secretas SSL y TLS

IBM WebSphere 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, o después de que el canal haya estado desocupado durante un período de tiempo.

El valor de restablecimiento de claves siempre lo establece el lado iniciador del canal 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.

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 entero en el rango comprendido entre 0 y 999999999, que representa el número de bytes no cifrados enviados y recibidos en una conversación SSL o TLS antes de que se renegocie la clave secreta SSL o TLS. Especificar un valor 0 indica que las claves secretas SSL o TLS no se renegocian nunca. Si especifica una cuenta de restablecimiento de clave secreta SSL o TLS entre 1 byte y 32 Kb, los canales SSL o 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 pequeñas SSL o TLS.

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 SSL o TLS en el archivo de configuración de cliente, consulte Stanza SSL del archivo de configuración de cliente.

Java

Para IBM WebSphere 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 sslReseto la propiedad de entorno MQC.SSL_RESET_COUNT_PROPERTY representa el número total de bytes enviados y recibidos por las clases WebSphere MQ para el código de cliente 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 también incluye la información de control enviada y recibida por las clases de WebSphere MQ para el cliente 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 WebSphere 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 WebSphere 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 SSL o TLS con una clave secreta que se renegocia después de que hayan fluido 4 MB de datos, emita el mandato siguiente para 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 SSL o TLS antes de que se renegocie la clave secreta.

Para obtener información sobre el uso de las propiedades del objeto en IBM WebSphere MQ classes for .NET, consulte Obtención y establecimiento de valores de atributo.

XMS .NET

Para clientes no gestionados XMS .NET, consulte Conexiones seguras a un gestor de colas de IBM WebSphere MQ.

Referencia relacionada

ALTER QMGR

DISPLAY QMGR

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 231.

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.

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 e utiliza SSL o TLS, la opción de CipherSpec determina el nivel de integridad de datos en la empresa. Si utiliza WebSphere 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.

Conexión de dos gestores de colas utilizando SSL o TLS

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 obtener y gestionar los certificados digitales. En un sistema de prueba, puede utilizar certificados o certificados autofirmados emitidos por una entidad emisora de certificados (CA) local. En un sistema de producción, no utilice certificados autofirmados. Para obtener más información, consulte ../zs14140_.dita.

Para obtener información detallada sobre la creación y la gestión de certificados, consulte “Trabajar con SSL o TLS en sistemas UNIX, Linux, and Windows” en la página 118.

Esta colección de temas presentan las tareas que forman parte de la configuración de las comunicaciones SSL y se proporciona una guía paso a paso sobre cómo 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 WebSphere MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

Notas:

1. En este contexto, un cliente SSL hace referencia a la conexión iniciando el reconocimiento.
2. Consulte el Glosario para obtener más detalles.

En sistemas UNIX, Linux y Windows, el cliente SSL o TLS envía un certificado sólo si tiene uno etiquetado en el formato WebSphere MQ correcto, que es `ibmwebspheremq` seguido por el nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`.

WebSphere 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 no se realiza correctamente sólo si el extremo del canal que actúa como el servidor SSL o TLS se ha definido con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información sobre la conexión de un gestor de colas de forma anónima, es decir, cuando el cliente SSL o TLS no envía un certificado, consulte "Conexión de dos gestores de colas utilizando autenticación unidireccional" en la página 216.

Etiquetas de certificados digitales, descripción de los requisitos

Al establecer SSL y 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.

Acerca de esta tarea

¿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 es independiente de los campos *Nombre distinguido del sujeto* o *Nombre común del sujeto* del certificado. Observe que *Nombre distinguido del sujeto* y *Nombre común del sujeto* son campos que están en el propio certificado. Se definen cuando se crea el certificado y no pueden cambiarse. No obstante, si fuera necesario, puede cambiar la etiqueta asociada con un certificado digital.

¿Cómo se utiliza la etiqueta de certificado?

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

Las etiquetas de certificado siguen un convenio de denominación; debe asegurarse de que utiliza el convenio de denominación de etiquetas correcto correspondiente a la plataforma que va a utilizar.

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

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 WebSphere MQ, el servidor SSL o 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 SSL o 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 está actuando como el servidor SSL o TLS está definido con el parámetro SSLCAUTH definido en un conjunto de valores de parámetro REQUIRED o SSLPEER.

Si desea más información sobre cómo conectarse a un gestor de colas utilizando la autenticación unidireccional, es decir, cuando un cliente SSL o TLS no envía un certificado, consulte "Conexión de dos gestores de colas utilizando autenticación unidireccional" en la página 216.

Sistemas , UNIX, Linux, and Windows

Acerca de esta tarea

En sistemas , UNIX, Linux, and Windows , el servidor SSL o TLS envía un certificado al cliente, sólo si el servidor encuentra uno etiquetado en el formato IBM WebSphere MQ correcto. En estos sistemas, el formato correcto es `ibmwebspheremq`, seguido del nombre del gestor de colas cambiado a minúsculas.

Por ejemplo, para un gestor de colas llamado `QM1`, el requisito de etiqueta de certificado es:

```
ibmwebspheremqqm1
```

Si no se ha encontrado ningún certificado en el repositorio de claves del gestor de colas que coincida con la etiqueta necesaria, con las letras minúsculas y el formato correcto, se produce un error y el reconocimiento SSL o TLS falla.

IBM WebSphere MQ cliente

Acerca de esta tarea

Al conectarse desde una aplicación cliente IBM WebSphere MQ, el cliente SSL o TLS sólo envía un certificado si tiene un certificado con una etiqueta con el formato `ibmwebspheremq`, seguido por el nombre de usuario del usuario que ejecuta el proceso de aplicación cliente.

Por ejemplo, para el nombre de usuario `wasadmin`, el requisito de etiqueta de certificado es como se muestra, doblado a minúsculas:

```
   ibmwebspheremqwasadmin
```

El requisito de etiqueta anterior se aplica a clientes de Message Service para C o C++ y .NET.

Cliente IBM WebSphere MQ Java o IBM WebSphere MQ JMS

Acerca de esta tarea

Los clientes IBM WebSphere MQ Java o IBM WebSphere MQ JMS utilizan los recursos de su proveedor JSSE (Java Secure Socket Extension) para seleccionar un certificado personal durante el reconocimiento SSL o TLS y, por lo tanto, no están sujetos a los requisitos de la etiqueta de certificado.

El comportamiento predeterminado es que el cliente JSSE examine los certificados del repositorio 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 resolver problemas o para entender mejor el reconocimiento que realiza la aplicación cliente IBM WebSphere MQ Java, en combinación con su proveedor JSSE específico, puede habilitar la depuración estableciendo

```
javax.net.debug=ssl
```

en el entorno de JVM.

Puede utilizar `-Djavax.net.debug=ssl` en la línea de mandatos o establecer la variable dentro de la aplicación, o a través de la configuración.

Conceptos relacionados

“Importación de un certificado personal en un repositorio de claves en sistemas UNIX, Linux, and Windows” en la página 138

Siga este procedimiento para importar un certificado personal.

Utilización de certificados autofirmados para la autenticación mutua de dos gestores de colas

Siga estas instrucciones de ejemplo para implementar la autenticación mutua entre dos gestores de colas, utilizando certificados SSL o TLS autofirmados.

Acerca de esta tarea

Escenario:

- Sean dos gestores de colas, QM1 y QM2, que deben comunicarse de forma segura. Se necesita una autenticación mutua entre QM1 y QM2.
- Se ha decidido probar la comunicación segura utilizando certificados autofirmados.

La configuración resultante es parecida a la siguiente:

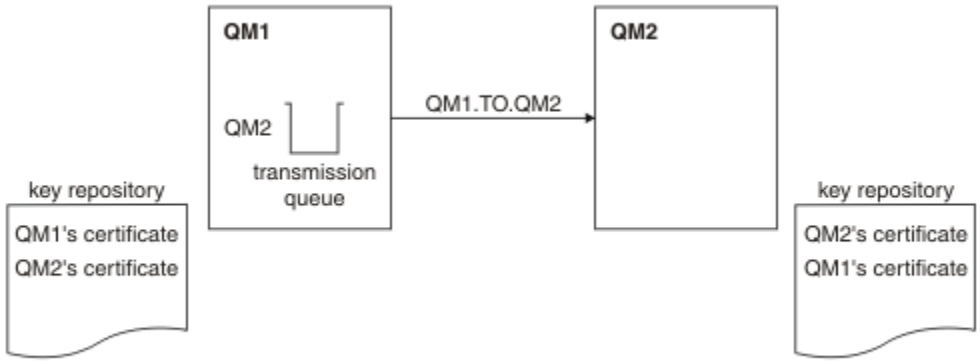


Figura 20. Configuración resultante de esta tarea

En Figura 14 en la página 213, el repositorio de claves para QM1 contiene el certificado para QM1 y el certificado público de QM2. El repositorio de claves para QM2 contiene el certificado para QM2 y el certificado público de QM1.

Procedimiento

1. Prepare el repositorio de claves en cada gestor de colas según el sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Cree un certificado autofirmado para cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
3. Extraiga una copia de cada certificado:
   - En sistemas UNIX, Linux y Windows.
4. Transfiera la parte pública del certificado de QM1 al sistema de QM2 y viceversa, utilizando un programa de utilidad como FTP.
5. Agregue el certificado de asociado al repositorio de claves por cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
6. En QM1, defina un canal emisor y la cola de transmisión asociada en el gestor de colas emitiendo mandatos como en el siguiente ejemplo:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QM1.MACH.COM) XMITQ(QM2)
SSLCIPH(RC4_MD5_US) DESCR('Sender channel using SSL from QM1 to QM2')
```

```
DEFINE QLOCAL(QM2) USAGE(XMITQ)
```

Este ejemplo utiliza la CipherSpec RC4_MD5. Las CipherSpecs de cada extremo del canal deben ser la misma.

7. En QM2, defina un canal receptor emitiendo un mandato como en el siguiente ejemplo:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from QM1 to QM2')
```

El canal debe tener el mismo nombre que el canal emisor que ha definido en el paso 6 y utilizar la misma CipherSpec.

8. Inicie el canal.

Resultados

Se crean los depósitos de claves y los canales, tal como se ilustra en la Figura 14 en la página 213

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante será similar a la mostrada en los ejemplos siguientes.

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM1.TO.QM2)                     CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=QM2,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

Desde el gestor de colas QM2, entre el siguiente mandato:

```
DISPLAY CHS(QM1.TO.QM2) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(QM1.TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(QM2.TO.QM1)                     CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QM1)
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QM1,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

En cada caso, el valor de SSLPEER debe coincidir con el del DN en el certificado de socio que se ha creado en el paso 2. El nombre del emisor coincide con el nombre de igual porque el certificado es autofirmado.

SSLPEER es opcional. Si se especifica, su valor debe establecerse para que se permita el DN del certificado de asociado (creado en el paso 2). Para obtener más información sobre el uso de SSLPEER, consulte WebSphere MQ rules for SSLPEER values .

Utilización de certificados autofirmados por CA para la autenticación mutua de dos gestores de colas

Siga estas instrucciones de ejemplo para implementar la autenticación mutua entre dos gestores de colas, utilizando certificados SSL o TLS firmados por una CA.

Acerca de esta tarea

Escenario:

- Dispone de dos gestores de colas denominados QMA y QMB, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre QMA y QMB.
- En el futuro está previsto utilizar esta red en un entorno de producción y, por consiguiente, se ha decidido utilizar certificados firmados por CA desde el principio.

La configuración resultante es parecida a la siguiente:

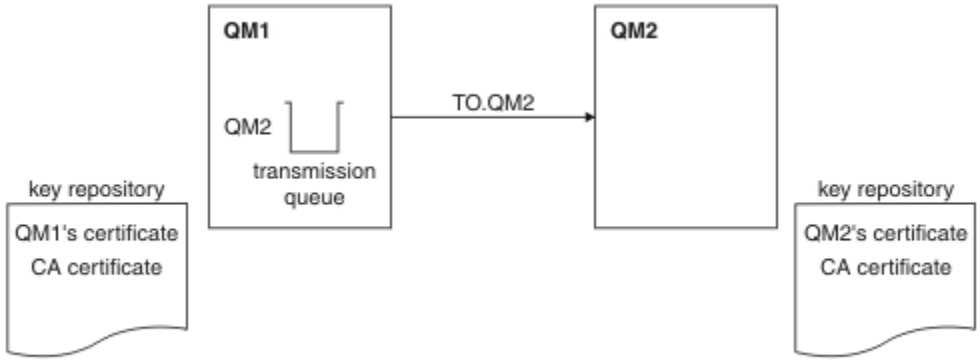


Figura 21. Configuración resultante de esta tarea

En la Figura 15 en la página 215, el repositorio de claves para QMA contiene el certificado de QMA y el certificado CA. El repositorio de claves para QMB contiene el certificado de QMB y el certificado CA. En este ejemplo, tanto el certificado de QMA como el certificado de QMB se emitieron mediante la misma CA (entidad emisora de certificados). Si el certificado de QMA y el certificado de QMB los emitieron CA diferentes, los repositorios de claves para QMA y QMB contendrán ambos certitificados de CA.

Procedimiento

1. Prepare el repositorio de claves en cada gestor de colas según el sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Solicite un certificado firmado por una CA para cada gestor de colas.

   Puede utilizar CA diferentes para los dos gestores de colas.
   - En sistemas UNIX, Linux y Windows.
3. Añada el certificado de una entidad emisora de certificados al repositorio de claves para cada gestor de colas:

   Si los gestores de colas utilizan entidades emisoras de certificados (CA) diferentes, el certificado de CA de cada entidad emisora de certificados deberá añadirse a ambos repositorios de claves.
   - En sistemas UNIX, Linux y Windows.
4. Añada el certificado firmado por la CA en el repositorio de claves para cada gestor de colas:
   - En sistemas UNIX, Linux y Windows.
5. En QMA, defina un canal emisor y la cola de transmisión asociada en el gestor de colas emitiendo mandatos como en el siguiente ejemplo:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(QMB.MACH.COM) XMITQ(QMB)
SSLCIPH(RC2_MD5_EXPORT) DESCR('Sender channel using SSL from QMA to QMB')

DEFINE QLOCAL(QMB) USAGE(XMITQ)
```

Este ejemplo utiliza la CipherSpec RC4_MD5. Las CipherSpecs de cada extremo del canal deben ser la misma.

6. En QMB, defina un canal receptor emitiendo un mandato como en el siguiente ejemplo:

```
DEFINE CHANNEL(TO.QMB) CHLTYPE(RCVR) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL to QMB')
```

El canal debe tener el mismo nombre que el canal emisor que ha definido en el paso 6 y utilizar la misma CipherSpec.

7. Inicie el canal:

Resultados

Se crean repositorios de claves y canales, como se ilustra en Figura 15 en la página 215.

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante es parecida a la que se muestra en los ejemplos siguientes.

Desde el gestor de colas QMA, entre el siguiente mandato:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QMB)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QMB)
```

Desde el gestor de colas QMB, entre el siguiente mandato:

```
DISPLAY CHS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QMB)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
   XMITQ( )
```

En cada caso, el valor de SSLPEER debe coincidir con el del nombre distinguido (DN) en el certificado de socio que se ha creado en el paso 2. El nombre del emisor coincide con el DN de sujeto del certificado de CA que ha firmado el certificado personal añadido en el Paso 4.

Conexión de dos gestores de colas utilizando autenticación unidireccional

Siga estas instrucciones de ejemplo para modificar un sistema con la autenticación mutua para permitir a un gestor de colas conectarse con otro utilizando la autenticación unidireccional; es decir, cuando el cliente SSL o TLS no envía un certificado.

Acerca de esta tarea

Escenario:

- Los dos gestores de colas (QM1 y QM2) se han configurado como en "Utilización de certificados autofirmados por CA para la autenticación mutua de dos gestores de colas" en la página 214.
- Desea cambiar QM1 para que se conecte a QM2 utilizando la autenticación unidireccional.

La configuración resultante es parecida a la siguiente:

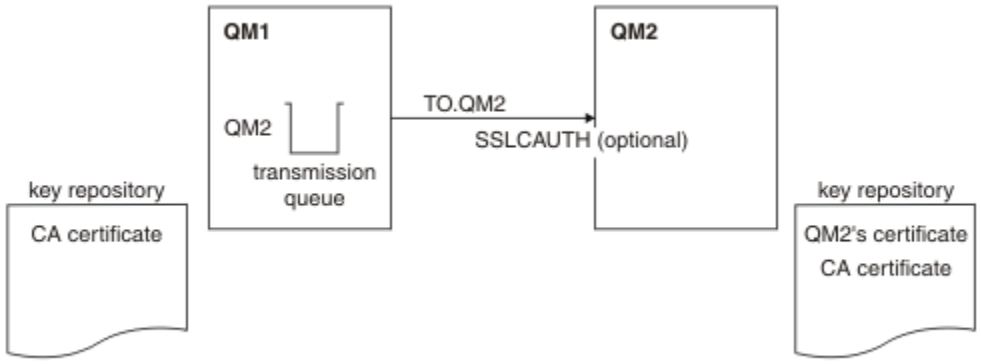


Figura 22. Gestores de colas que permiten la autenticación unidireccional

Procedimiento

1. Elimine el certificado personal de QM1 de su repositorio de claves, de acuerdo con el sistema operativo:
   - En los sistemas UNIX, Linux y Windows. El certificado se etiqueta como se indica a continuación:
     - `ibmwebspheremq` seguido del nombre del gestor de colas en minúsculas. Por ejemplo, para `QM1`, `ibmwebspheremqqm1`.
2. Opcional: En QM1, si algún canal SSL o TLS se ha ejecutado anteriormente, renueve el entorno SSL o TLS.
3. Permitir conexiones anónimas en el receptor.

Resultados

Los repositorios de claves y los canales cambian como se ilustra en Figura 16 en la página 217.

Qué hacer a continuación

Si el canal emisor estaba en ejecución y ha emitido el mandato REFRESH SECURITY TYPE(SSL) (en el paso 2), el canal se reiniciará automáticamente. Si el canal emisor no estaba en ejecución, inícielo.

En el extremo del canal del servidor, la presencia del valor de parámetro de nombre de igual en la visualización del estado del canal indica que se ha emitido un certificado de cliente.

Compruebe que la tarea se ha completado satisfactoriamente emitiendo algunos mandatos DISPLAY. Si la tarea se ha realizado satisfactoriamente, la salida resultante es similar a la que se muestra en los ejemplos siguientes:

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
     4 : DISPLAY CHSTATUS(TO.QMB) SSLPEER
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(SDR)
   CONNAME(9.20.25.40)                     CURRENT
   RQMNAME(QM2)
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMB,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(MQGET)
   XMITQ(QM2)
```

Desde el gestor de colas QM2 ejecute el siguiente mandato:

```
DISPLAY CHS(TO.QM2) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(TO.QM2) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(TO.QM2)                         CHLTYPE(RCVR)
   CONNAME(9.20.35.92)                     CURRENT
   RQMNAME(QMA)                            SSLCERTI( )
   SSLPEER( )                              STATUS(RUNNING)
   SUBSTATE(RECEIVE)                       XMITQ( )
```

En QM2, el campo SSLPEER está vacío, lo que muestra que QM1 no ha enviado un certificado. En QM1, el valor de SSLPEER coincide con el del DN del certificado personal de QM2.

Conexión de un cliente a un gestor de colas de forma segura

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 obtener y gestionar los certificados digitales. En un sistema de prueba, puede utilizar certificados o certificados autofirmados emitidos por una entidad emisora de certificados (CA) local. En un sistema de producción, no utilice certificados autofirmados. Para obtener más información, consulte ../zs14140_.dita.

Para obtener información detallada sobre la creación y la gestión de certificados, consulte “Trabajar con SSL o TLS en sistemas UNIX, Linux, and Windows” en la página 118.

Esta colección de temas presentan las tareas que forman parte de la configuración de las comunicaciones SSL y se proporciona una guía paso a paso sobre cómo 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 WebSphere MQ, el servidor SSL o TLS siempre solicita un certificado del cliente.

En sistemas UNIX, Linux, and Windows, el cliente SSL o TLS envía un certificado sólo si tiene uno etiquetado en el formato WebSphere MQ correcto, que es `ibmwebspheremq` seguido por el ID de usuario de inicio de sesión en minúsculas, por ejemplo `ibmwebspheremqmyuserid`.

WebSphere 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 no se realiza correctamente sólo si el extremo del canal que actúa como el servidor SSL o TLS se ha definido con el parámetro SSLCAUTH establecido en REQUIRED o un valor de parámetro SSLPEER establecido. Para obtener más información sobre la conexión de un gestor de colas de forma anónima, consulte “Conexión de un cliente a un gestor de colas de forma anónima” en la página 222.

Utilización de certificados autofirmados para la autenticación mutua de un cliente y un gestor de colas

Siga las instrucciones de este ejemplo para implementar la autenticación mutua entre un cliente y un gestor de colas, utilizando certificados SSL o TLS autofirmados.

Acerca de esta tarea

Escenario:

- Tiene un cliente, C1, y un gestor de colas, QM1, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre C1 y QM1.
- Ha decidido probar la comunicación segura utilizando certificados autofirmados.

DCM en IBM i no admite certificados autofirmados, por lo que esta tarea no se aplica en sistemas IBM i.

La configuración resultante es parecida a la siguiente:

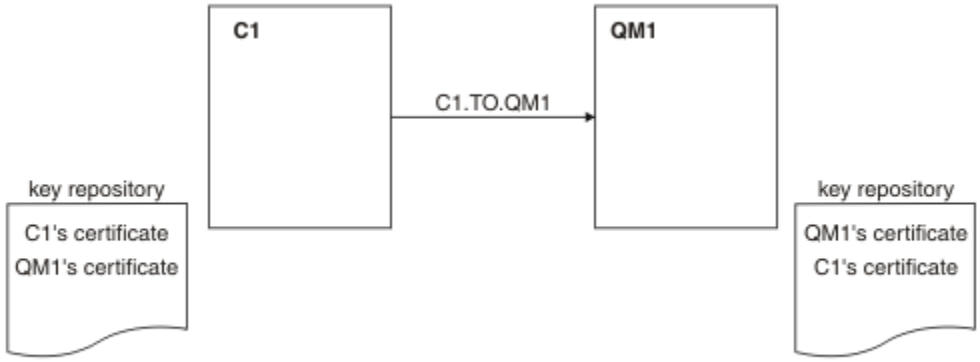


Figura 23. Configuración resultante de esta tarea

En Figura 17 en la página 219, el repositorio de claves para QM1 contiene el certificado para QM1 y el certificado público para C1. El repositorio de claves para C1 contiene el certificado para C1 y el certificado público de QM1.

Procedimiento

1. Prepare el repositorio de claves en el cliente y el gestor de colas conforme al sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Cree certificados autofirmados para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
3. Extraiga una copia de cada certificado:
   - En sistemas UNIX, Linux y Windows.

4. Transfiera la parte pública del certificado de C1 al sistema de QM1 y viceversa, utilizando un programa de utilidad como FTP.
5. Añada el certificado de socio al repositorio de claves para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
6. Emita el mandato REFRESH SECURITY TYPE (SSL) en el gestor de colas.
7. Defina un canal de conexión con el cliente siguiendo uno de estos métodos:
   - Utilizando la llamada MQCONNX con la estructura MQSCO en C1, tal como se describe en Creación de un canal de conexión-cliente en el cliente MQI de WebSphere MQ.
   - Utilizando una tabla de definiciones de canal de cliente, tal como se describe en Creación de definiciones de conexión de servidor y conexión de cliente en el servidor .
8. En QM1, defina un canal de conexión con el servidor, utilizando un mandato como el del ejemplo siguiente:

   ```
   DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC4_MD5_US)
   SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
   ```

   El canal debe tener el mismo nombre que el canal de conexión de cliente que ha definido en el paso 6, y utilizar la misma especificación de cifrado (CipherSpec).

Resultados

Se crean los repositorios de claves y los canales, tal como se ilustra en la Figura 17 en la página 219

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida del resultado es parecida a la que se muestra en el siguiente ejemplo.

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                       CURRENT
   SSLCERTI("CN=QM1,OU=WebSphere MQ Development,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5E:02,CN=QM2,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                           SUBSTATE(RECEIVE)
```

Es opcional establecer el atributo de filtro SSLPEER de las definiciones de canal. Si se establece la definición de canal SSLPEER, su valor debe coincidir con el DN de sujeto en el certificado de socio que se ha creado en el paso 2. Después de una conexión satisfactoria, el campo SSLPEER de la salida DISPLAY CHSTATUS muestra el DN de asunto del certificado de cliente remoto.

Utilización de certificados firmados por CA para la autenticación mutua de un cliente y un gestor de colas

Siga las instrucciones de este ejemplo para implementar la autenticación mutua entre un cliente y un gestor de colas, utilizando certificados SSL o TLS firmados por CA.

Acerca de esta tarea

Escenario:

- Tiene un cliente, C1, y un gestor de colas, QM1, que deben comunicarse de forma segura. Necesita que se lleve a cabo autenticación mutua entre C1 y QM1.
- En el futuro está previsto utilizar esta red en un entorno de producción y, por consiguiente, se ha decidido utilizar certificados firmados por CA desde el principio.

La configuración resultante es parecida a la siguiente:

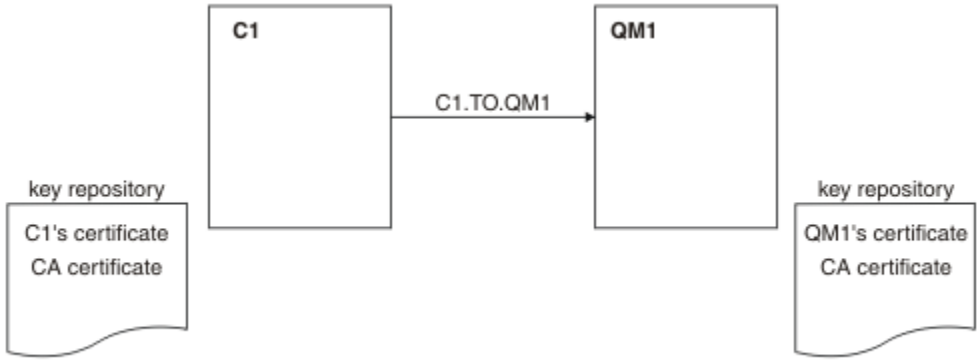


Figura 24. Configuración resultante de esta tarea

En Figura 18 en la página 221, el repositorio de claves para C1 contiene el certificado para C1 y el certificado de CA. El repositorio de claves de QM1 contiene el certificado de QM1 y el certificado de la CA. En este ejemplo, tanto el certificado de C1 como el certificado de QM1 se emitieron mediante la misma CA (entidad emisora de certificados). Si el certificado de C1 y el certificado de QM1 los emitieron CA diferentes, los repositorios de claves para C1 y QM1 contendrán ambos certificados de CA.

Procedimiento

1. Prepare el repositorio de claves en el cliente y el gestor de colas conforme al sistema operativo:
   - En sistemas UNIX, Linux y Windows.
2. Solicite un certificado firmado por CA para el cliente y el gestor de colas.

   Puede utilizar autoridades emisoras de certificados (CA) distintas para el cliente y el gestor de colas.
   - En sistemas UNIX, Linux y Windows.
3. Añada el certificado de la entidad emisora de certificados al repositorio de claves para el cliente y el gestor de colas.

   Si el cliente y los gestores de colas utilizan entidades emisoras de certificados (CA) diferentes, el certificado de CA de cada entidad emisora de certificados debe añadirse a ambos repositorios de claves.
   - En sistemas UNIX, Linux y Windows.
4. Añada el certificado firmado por CA al repositorio de claves para el cliente y el gestor de colas:
   - En sistemas UNIX, Linux y Windows.
5. Defina un canal de conexión con el cliente siguiendo uno de estos métodos:
   - Utilizando la llamada MQCONNX con la estructura MQSCO en C1, tal como se describe en Creación de un canal de conexión-cliente en el cliente MQI de WebSphere MQ.
   - Utilizando una tabla de definiciones de canal de cliente, tal como se describe en Creación de definiciones de conexión de servidor y conexión de cliente en el servidor .
6. En QM1, defina un canal de conexión con el servidor emitiendo un mandato como el del ejemplo siguiente:

```
DEFINE CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) TRPTYPE(TCP) SSLCIPH(RC2_MD5_EXPORT)
SSLCAUTH(REQUIRED) DESCR('Receiver channel using SSL from C1 to QM1')
```

El canal debe tener el mismo nombre que el canal de conexión de cliente que ha definido en el paso 6, y utilizar la misma especificación de cifrado (CipherSpec).

Resultados

Se crean repositorios de claves y canales, como se ilustra en Figura 18 en la página 221.

Qué hacer a continuación

Compruebe que la tarea ha finalizado satisfactoriamente utilizando mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida resultante es parecida a la que se muestra en el siguiente ejemplo.

Desde el gestor de colas QM1, especifique el siguiente mandato:

```
DISPLAY CHSTATUS(TO.QMB) SSLPEER SSLCERTI
```

La salida resultante es como la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI("CN=WebSphere MQ CA,OU=WebSphere MQ Devt,O=IBM,ST=Hampshire,C=UK")
   SSLPEER("SERIALNUMBER=4C:D0:49:D5:02:5F:38,CN=QMA,OU=WebSphere MQ
Development,O=IBM,ST=Hampshire,C=UK")
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

El campo SSLPEER de la salida DISPLAY CHSTATUS muestra el DN de sujeto del certificado de cliente remoto que se ha creado en el Paso 2. El nombre del emisor coincide con el DN de sujeto del certificado de CA que ha firmado el certificado personal añadido en el Paso 4.

Conexión de un cliente a un gestor de colas de forma anónima

Siga las instrucciones de este ejemplo para modificar un sistema con autenticación mutua a fin de permitir que un gestor de colas se conecte a otro de forma anónima.

Acerca de esta tarea

Escenario:

- Su gestor de colas y cliente (QM1 y C1) se han configurado como se indica en “Utilización de certificados firmados por CA para la autenticación mutua de un cliente y un gestor de colas” en la página 220.
- Desea cambiar C1 de modo que se conecte de forma anónima a QM1.

La configuración resultante es parecida a la siguiente:

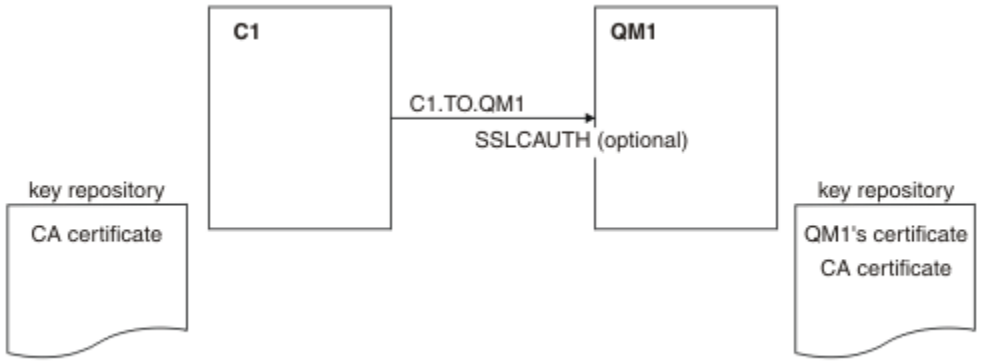


Figura 25. Cliente y gestor de colas que permiten conexión anónima

Procedimiento

1. Elimine el certificado personal del repositorio de claves para C1, conforme al sistema operativo:
   - En los sistemas UNIX, Linux y Windows. El certificado se etiqueta como se indica a continuación:
     - `ibmwebspheremq` seguido por el ID de usuario de inicio de sesión doblado a minúsculas, por ejemplo `ibmwebspheremqmyuserid`.
2. Reinicie la aplicación cliente o haga que la aplicación cliente se cierre y abra de nuevo todas las conexiones SSL o TLS.
3. Permita las conexiones anónimas en el gestor de colas emitiendo el siguiente mandato:

```
ALTER CHANNEL(C1.TO.QM1) CHLTYPE(SVRCONN) SSLCAUTH(OPTIONAL)
```

Resultados

Los repositorios de claves y los canales cambian como se ilustra en Figura 19 en la página 222.

Qué hacer a continuación

En el extremo del canal del servidor, la presencia del valor de parámetro de nombre de igual en la visualización del estado del canal indica que se ha emitido un certificado de cliente.

Compruebe que la tarea se ha completado satisfactoriamente emitiendo algunos mandatos DISPLAY. Si la tarea ha sido satisfactoria, la salida del resultado es parecida a la que se muestra en el siguiente ejemplo:

Desde el gestor de colas QM1, ejecute el siguiente mandato:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
```

La salida resultante será parecida a la del ejemplo siguiente:

```
DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
     5 : DISPLAY CHSTATUS(C1.TO.QM1) SSLPEER SSLCERTI
AMQ8417: Display Channel Status details.
   CHANNEL(C1.TO.QM1)                      CHLTYPE(SVRCONN)
   CONNAME(9.20.35.92)                     CURRENT
   SSLCERTI( )                             SSLPEER( )
   STATUS(RUNNING)                         SUBSTATE(RECEIVE)
```

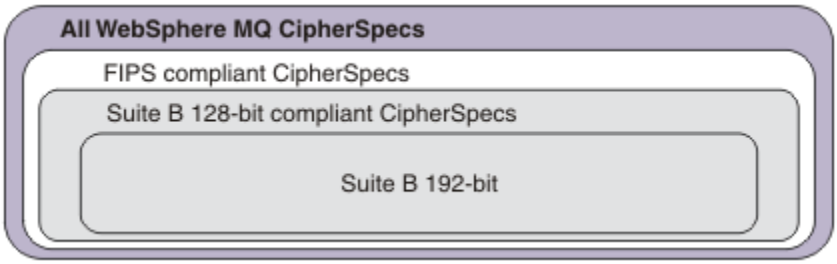
Los campos SSLCERTI y SSLPEER están vacíos, lo que indica que C1 no ha enviado ningún certificado.

Especificación de CipherSpecs

Especifique una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC de **DEFINE CHANNEL** o en el mandato MQSC de **ALTER CHANNEL** .

Algunas de las CipherSpecs que puede utilizar con IBM WebSphere MQ son compatibles con FIPS. Otras, como por ejemplo, NULL_MD5 no lo son. Asimismo, algunas de las CipherSpecs compatibles con FIPS también son compatibles con Suite B, aunque otras 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:



Las especificaciones de cifrado que puede utilizar con el soporte de SSL y TLS de IBM WebSphere MQ aparecen listadas en la tabla siguiente. 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 SSL es el tamaño almacenado en el certificado, a menos que esté determinado por la CipherSpec, tal como está indicado en la tabla.

Nombre de CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de cifrado	Bits de cifrado	FIPS[1]	Suite B de 128 bits	Suite B de 192 bits
NULL_MD5 [a]	SSL 3.0	MD5	Ninguna	0	No	No	No
NULL_SHA [a]	SSL 3.0	SHA-1	Ninguna	0	No	No	No
RC4_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC4	40	No	No	No
RC4_MD5_US [a]	SSL 3.0	MD5	RC4	128	No	No	No
RC4_SHA_US [a]	SSL 3.0	SHA-1	RC4	128	No	No	No
RC2_MD5_EXPORT [2] [a]	SSL 3.0	MD5	RC2	40	No	No	No
DES_SHA_EXPORT [2] [a]	SSL 3.0	SHA-1	DES	56	No	No	No
RC4_56_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	RC4	56	No	No	No
DES_SHA_EXPORT1024 [3] [b]	SSL 3.0	SHA-1	DES	56	No	No	No
TLS_RSA_WITH_AES_128_CBC_SHA [a]	TLS 1.0	SHA-1	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_CBC_SHA [4] [a]	TLS 1.0	SHA-1	AES	256	Sí	No	No
TLS_RSA_WITH_DES_CBC_SHA [a]	TLS 1.0	SHA-1	DES	56	No[5]	No	No
FIPS_WITH_DES_CBC_SHA [b]	SSL 3.0	SHA-1	DES	56	No[6]	No	No
TLS_RSA_WITH_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	No
TLS_RSA_WITH_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
TLS_RSA_WITH_AES_256_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	256	Sí	No	No
ECDHE_ECDSA_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No
ECDHE_RSA_RC4_128_SHA256 [b]	TLS 1.2	SHA_1	RC4	128	No	No	No
ECDHE_ECDSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
ECDHE_ECDSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sí	No	No

Nombre de CipherSpec	Protocolo utilizado	Algoritmo MAC	Algoritmo de cifrado	Bits de cifrado	FIPS[1]	Suite B de 128 bits	Suite B de 192 bits
ECDHE_RSA_AES_128_CBC_SHA256 [b]	TLS 1.2	SHA-256	AES	128	Sí	No	No
ECDHE_RSA_AES_256_CBC_SHA384 [b]	TLS 1.2	SHA-384	AES	256	Sí	No	No
ECDHE_ECDSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	Sí	No
ECDHE_ECDSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	Sí
ECDHE_RSA_AES_128_GCM_SHA256 [b]	TLS 1.2	AEAD AES-128 GCM	AES	128	Sí	No	No
ECDHE_RSA_AES_256_GCM_SHA384 [b]	TLS 1.2	AEAD AES-256 GCM	AES	256	Sí	No	No
TLS_RSA_WITH_NULL_SHA256 [b]	TLS 1.2	SHA-256	Ninguna	0	No	No	No
ECDHE_RSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Ninguna	0	No	No	No
ECDHE_ECDSA_NULL_SHA256 [b]	TLS 1.2	SHA-1	Ninguna	0	No	No	No
TLS_RSA_WITH_NULL_NULL [b]	TLS 1.2	Ninguna	Ninguna	0	No	No	No
TLS_RSA_WITH_RC4_128_SHA256 [b]	TLS 1.2	SHA-1	RC4	128	No	No	No

Notas:

1. 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.
2. 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.
3. El tamaño de clave de reconocimiento es de 1024 bits.
4. Este CipherSpec no se puede utilizar para garantizar una conexión desde WebSphere MQ Explorer a un gestor de colas amenos que se apliquen los archivos de políticas no restringidas apropiados al JRE utilizado por Explorer.
5. Esta CipherSpec obtuvo el certificado FIPS 140-2 antes del 19 mayo de 2007.
6. 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.
7. Se puede utilizar esta CipherSpec para transferir hasta 32 GB de datos antes de que la conexión concluya con el error AMQ9288. Para evitar este error, evite utilizar triple DES, o habilite el restablecimiento de clave secreta cuando utilice esta CipherSpec.

Soporte de plataformas:

- a Disponible en todas las plataformas soportadas.
- b Disponible sólo en plataformas UNIX, Linux, and Windows.

Conceptos relacionados

"Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ" en la página 36
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 WebSphere MQ.

Referencia relacionada

DEFINE CHANNEL

ALTER CHANNEL

Obtención de información sobre CipherSpecs utilizando IBM WebSphere MQ Explorer

Puede utilizar IBM WebSphere MQ Explorer para visualizar descripciones de CipherSpecs.

Utilice el procedimiento siguiente para obtener información acerca de las CipherSpecs que aparecen en la "Especificación de CipherSpecs" en la página 223:

1. Abra **IBM WebSphere 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 SSL, 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 WebSphere 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 SSL, es posible que el sistema dé soporte a nuevas CipherSpecs que no figuran en la "Especificación de CipherSpecs" en la página 223. 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 CipherSpec de SSL que sea válida y que esté soportada por la versión de SSL en la que se esté ejecutando el sistema.

IBM i

Una serie de dos caracteres que representa un valor hexadecimal.

Para obtener más información sobre los valores permitidos, consulte la documentación del producto adecuada (busque *cipher_spec* en la documentación del productoIBM i).

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 dos caracteres que representa un valor hexadecimal. Los códigos hexadecimales se corresponden con los valores definidos en el protocolo SSL.

Para obtener más información, consulte la descripción de gsk_environment_open() en el capítulo de consultas de API de la publicación *z/OS Cryptographic Services System SSL Programming*, SC24-5901,

donde hay una lista de todas las especificaciones de cifrado SSL V3.0 y TLS V1.0 soportadas en formato de códigos hexadecimales de dos dígitos.

Consideraciones para los clústeres de WebSphere MQ

Con los clústeres de WebSphere MQ, es más seguro utilizar los nombres de CipherSpec de la "Especificación de CipherSpecs" en la página 223. Si utiliza una especificación alternativa, tenga en cuenta que la especificación puede no ser válida en otras plataformas. Para obtener más información, consulte "SSL y clústeres" en la página 254.

Especificación de una CipherSpec para un cliente MQI de IBM WebSphere MQ

Tiene tres opciones para especificar una CipherSpec para un cliente MQI de IBM WebSphere MQ .

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 clases IBM WebSphere MQ para Java y clases IBM WebSphere MQ para JMS

Las clases IBM WebSphere MQ para Java y las clases IBM WebSphere MQ para JMS especifican CipherSuites de forma diferente a otras plataformas.

Para obtener información sobre cómo especificar una CipherSuite con clases IBM WebSphere MQ para Java, consulte Soporte SSL (Secure Sockets Layer).

Para obtener información sobre cómo especificar una CipherSuite con clases IBM WebSphere MQ para JMS, consulte Utilización de SSL (Secure Sockets Layer) con WebSphere MQ classes for JMS.

Auditoría

Puede comprobar las intrusiones de seguridad, o intentos de intrusión, mediante mensajes de sucesos. También puede comprobar la seguridad del sistema utilizando IBM WebSphere 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 SSL 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 accesos de WebSphere 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 WebSphere MQ para 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 (sólo 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 (sólo 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 la plataforma, autorice el acceso para conectarse al gestor de colas y para transferir la cola a dicho gestor 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 los sistemas UNIX, Linux y Windows , emita los mandatos siguientes:

```
setmqaut -m QMgrName -t qmgr -g GroupName +connect
setmqaut -m QMgrName -t queue -n QueueName -g GroupName -all +put
```

  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.

  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 198.

Información relacionada

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 SSL/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.

   Windows UNIX Linux **Windows, sistemas 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.

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)
```

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 y clústeres

Al configurar SSL 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 SSL, debe configurar SSL en todos los gestores de colas que se comuniquen utilizando el canal.

Para obtener más información sobre SSL, consulte Soporte de WebSphere MQ para SSL y TLS. 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 WebSphere 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 SSL, se aplican las siguientes consideraciones:

- Todos los gestores de colas que deseen comunicarse con este CLUSRCVR debe tener acceso al soporte de SSL. Esta provisión de SSL 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 WebSphere 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=WebSphere MQ Development, CN=DEVT.QM*` en el CLUSRCVR permitirá que todos los canales de clúster emisor se conecten correctamente, pero impedirá que se conecten canales de clúster emisor 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 SSL 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 RC4_MD5_US).

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 agrupados en clúster a SSL

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 WebSphere 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.

Actualice todos los gestores de colas del clúster a WebSphere MQ V6 o superior, si aún no están en estos niveles. Distribuya los certificados y las claves para que SSL funcione desde cada uno de ellos.

Acerca de esta tarea

Cambie los canales CLUSRCVR de uno en uno, y permita que los cambios circulen por el clúster antes de cambiar el siguiente. Asegúrese de no cambiar la ruta inversa hasta que los cambios para el canal actual se hayan distribuido por el clúster.

Procedimiento

1. Cambie los canales CLUSRCVR a SSL en el orden que desee.

   Los cambios fluyen en la dirección opuesta por canales que no se han cambiado a SSL.
2. Cambie todos los canales CLUSSDR manuales a SSL.

   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 grandes, el uso del mandato **REFRESH CLUSTER** puede ser perjudicial para el clúster mientras está en curso y, también en intervalos de 27 días trascurridos los cuales los objetos del clúster envían automáticamente actualizaciones de estado a todos los gestores de colas. Consulte La renovación en un clúster grande puede afectar el rendimiento y la disponibilidad del clúster.

Conceptos relacionados

"Especificación de CipherSpecs" en la página 223
Especifique una CipherSpec utilizando el parámetro **SSLCIPH** en el mandato MQSC de **DEFINE CHANNEL** o en el mandato MQSC de **ALTER CHANNEL** .

"Certificados digitales y compatibilidad de CipherSpec en IBM WebSphere MQ" en la página 36
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 WebSphere MQ.

Información relacionada

Agrupación en clúster: utilización de las recomendaciones de REFRESH CLUSTER

Inhabilitación de SSL o TLS en gestores de colas y canales agrupados en clúster

Para desactivar SSL o 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. Establezca el valor del parámetro SSLCIPH en ' ', una serie vacía entre comillas simples .

   Puede desactivar SSL o TLS en los canales de clúster receptores en el orden que desee.

   Tenga en cuenta que los cambios fluyen en la dirección opuesta sobre los canales en los que deja SSL o 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 SSL o 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 26 en la página 257 y se describen en el siguiente ejemplo.

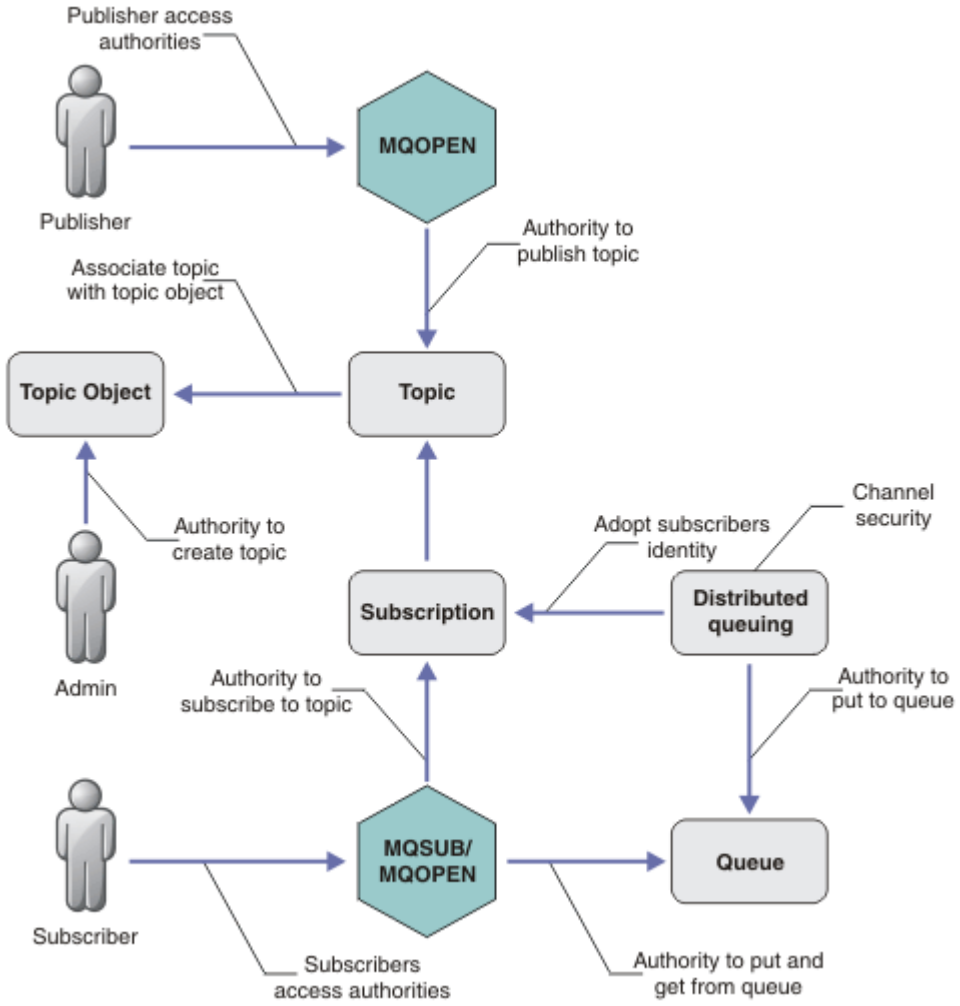


Figura 26. Relaciones de seguridad de publicación/suscripción

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. "Modelo de seguridad de temas" en la página 259 explica cómo proteger los temas utilizando 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 264 y en "Otorgar acceso a un usuario para publicar en un tema" en la página 269.

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 260

Suscripción mediante un tema
"Suscripción utilizando una serie del tema donde el nodo de tema no existe" en la página 261

Suscripción mediante un tema con comodines
"Suscripción utilizando una serie de tema que contiene caracteres comodín" en la página 262

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 274.

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 262.

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

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 WebSphere MQ, los canales se configuran con PUTAUT=DEF para colocar mensajes en las colas de destino mediante la autorización del proceso del canal. Puede modificar la configuración del canal para utilizar PUTAUT=CTX para exigir que el usuario que se suscribe tenga autorización para propagar una suscripción a otro gestor de colas en un clúster.

En la sección Seguridad de publicación/suscripción entre gestores de colas 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 WebSphere 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.

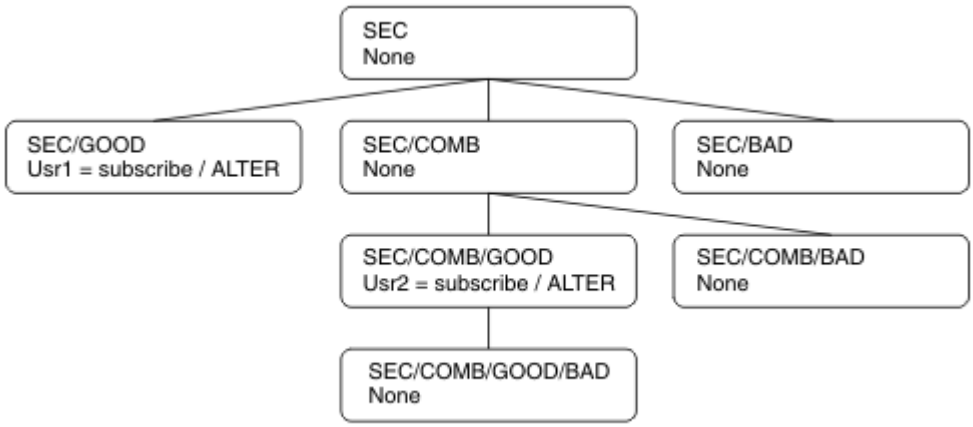
Los atributos de seguridad son una lista de control de acceso que indica qué autorización tiene un ID de usuario o grupo de seguridad determinado del sistema operativo sobre el objeto de tema.

Considere el ejemplo siguiente donde los objetos de tema se han definido con atributos de seguridad o autorizaciones:

Tabla 17. Ejemplo de autorizaciones de objetos de tema

Nombre de tema	Serie de tema	Autorizaciones - no z/OS	Autorizaciones z/OS
SECROOT	SEC	Ninguna	Ninguna
SECGOOD	SEC/GOOD	usr1+subscribe	ALTER HLQ.SUBSCRIBE.SECGOOD
SECBAD	SEC/BAD	Ninguna	Ninguna HLQ.SUBSCRIBE.SECBAD
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.

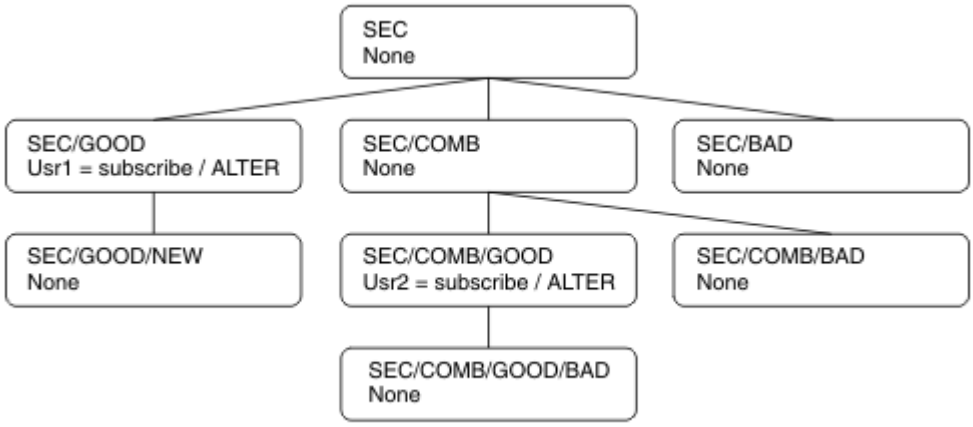
Suscripción utilizando una serie del tema donde el nodo de tema existe

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, *no* se realizan comprobaciones 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 18 en la página 264:

Tabla 18. 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 administrador
	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 caso de ejemplo que tiene la configuración de control de accesos sobre los temas de un modo que permite que el control de seguridad se aplique 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.

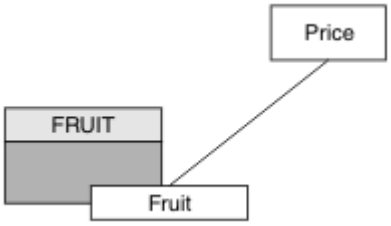


Figura 27. Ejemplo de acceso a objeto de tema

Tabla 19. 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:
   - 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:

     
     Windows, sistemas UNIX and Linux

     ```
     setmqaut -t topic -n FRUIT -p USER1 +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:

- 
  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"
```

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 264.

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.

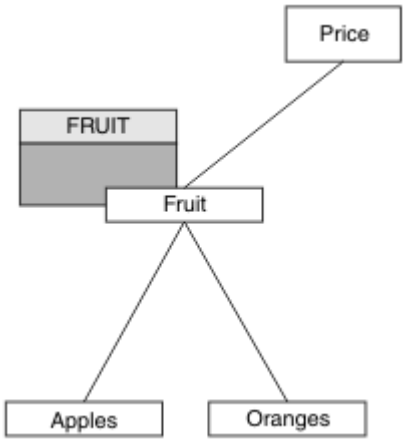


Figura 28. Ejemplo de otorgar acceso a un tema dentro de un árbol de temas

Tabla 20. 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 265.

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.

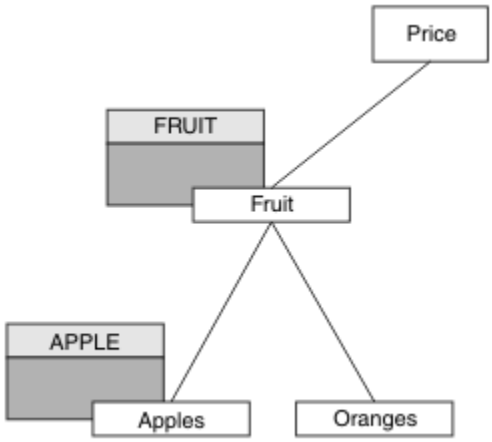


Figura 29. Otorgar acceso a temas específicos dentro de un árbol de temas

Tabla 21. 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:
   - Otras plataformas:

En la tarea anterior se otorgó acceso a USER1 para suscribirse al tema "Price/Fruit/Apples" otorgando acceso de suscripción al usuario al perfil FRUIT.

Este único perfil también ha otorgado 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 acceso de suscripción al usuario al perfil APPLE. Hágalo mediante el mandato de autorización para la plataforma:

Windows, sistemas UNIX and Linux

```
setmqaut -t topic -n APPLE -p USER2 +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:

- 

En plataformas Windows, UNIX y 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"
```

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 267 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.

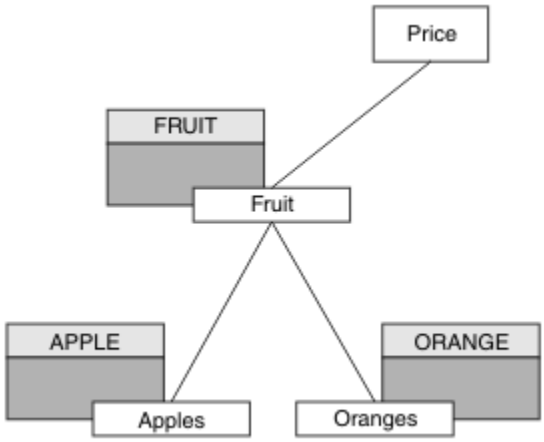


Figura 30. Ejemplo de otorgar control de acceso para evitar mensajes adicionales.

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:
   - Otras plataformas:

     Configure el acceso equivalente mediante los mandatos de autorización para la plataforma:

     Windows UNIX Linux **Windows, sistemas UNIX and Linux**

     ```
     setmqaut -t topic -n ORANGE -p USER1 +sub
     setmqaut -t topic -n APPLE -p USER1 +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:

- Windows UNIX Linux 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"
```

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:

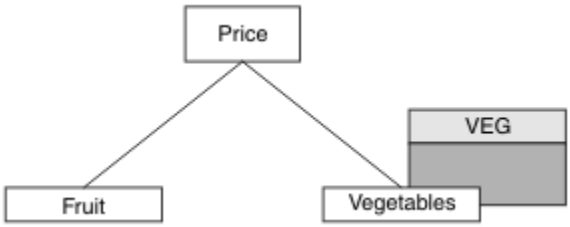


Figura 31. Otorgar acceso de publicación a un tema

Tabla 22. Ejemplo de requisitos de acceso de publicación

Tema	Acceso de publicación necesario	Objeto de tema
Price	Ningún usuario	Ninguna
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:
   - 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:

     Windows UNIX Linux **Windows, sistemas UNIX and Linux**

     ```
     setmqaut -t topic -n VEG -p USER1 +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:

- Windows UNIX Linux 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"
```

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 269.

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.

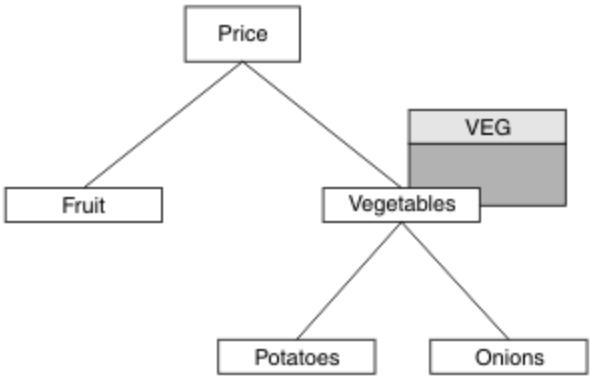


Figura 32. Otorgar acceso de publicación a un tema dentro de un árbol de temas

Tabla 23. 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 271.

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.

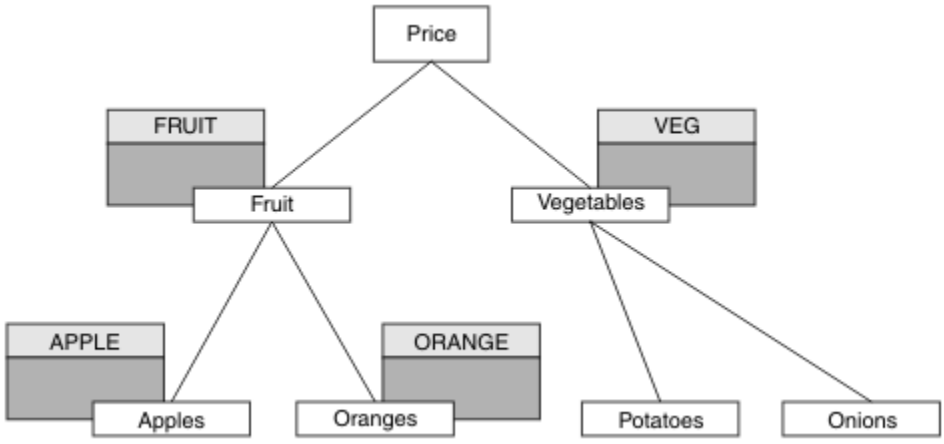


Figura 33. Otorgar acceso para publicar y suscribir

Tabla 24. 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:

- 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:

  Windows UNIX Linux **Windows, sistemas UNIX and Linux**

  ```
  setmqaut -t topic -n FRUIT -p USER1 +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:

- Windows UNIX Linux En plataformas Windows, UNIX y Linux , el suceso de autorización siguiente:

  ```
  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 25. 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

Tabla 25. 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/Fruit/Oranges	USER1		ORANGE
Price/Vegetables		USER1	VEG
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 26. 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.

IBM WebSphere MQ Advanced Message Security

IBM WebSphere MQ Advanced Message Security (AMS) es un componente con licencia separada de IBM WebSphere MQ Advanced Message Security que proporciona un nivel alto de protección para los datos

sensibles que fluyen por la red IBM WebSphere MQ Advanced Message Security, aunque no afecta a las aplicaciones finales.

IBM WebSphere MQ Advanced Message SecurityVisión general de

Las aplicaciones de IBM WebSphere MQ pueden utilizar IBM WebSphere MQ 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 IBM WebSphere MQ AMS

Comportamiento que ha cambiado entre la versión 7.0.1 y la versión 7.5

A medida que IBM Advanced Message Security se ha convertido en un componente en WebSphere MQ 7.5, algunos aspectos de la funcionalidad de IBM WebSphere MQ AMS han cambiado, lo que podría afectar a las aplicaciones existentes, los scripts administrativos o los procedimientos de gestión.

Revise la lista de cambios siguientes detenidamente antes de actualizar los gestores de colas a la versión 7.5. Determine si debe planificar realizar cambios en las aplicaciones, scripts y procedimientos existentes antes de empezar a migrar los sistemas a IBM WebSphere MQ versión 7.5:

- La instalación de IBM WebSphere MQ AMS es parte del proceso de instalación de WebSphere MQ.
- Las prestaciones de seguridad de IBM WebSphere MQ AMS se habilitan con su instalación y se controlan con políticas de seguridad. No es necesario habilitar los interceptores para permitir que IBM WebSphere MQ AMS comience a interceptar los datos.
- IBM WebSphere MQ AMS en WebSphere MQ versión 7.5 no requiere el uso del mandato **cfgmqs** como en la versión autónoma de IBM WebSphere MQ AMS.

Características y funciones de IBM WebSphere MQ Advanced Message Security

Advanced Message Security amplía los servicios de seguridad de WebSphere 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, IBM WebSphere MQ AMS verifica que el emisor de los datos de un mensaje está autorizado para colocar mensajes firmados en una cola de destino.

A continuación, se muestra una lista completa de funciones de IBM WebSphere MQ AMS:

- Protege las transacciones sensibles o de alto valor procesadas por WebSphere 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 aplicaciones existentes y aplicaciones escritas por el usuario para WebSphere MQ.

Tratamiento de errores

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 WebSphere MQ 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.

WebSphere MQ AMS utiliza SYSTEM.PROTECTION.ERROR.QUEUE como cola de manejo de errores. Todos los mensajes colocados por IBM WebSphere MQ AMS en SYSTEM.PROTECTION.ERROR.QUEUE están precedidos por la cabecera MQDLH.

El administrador de WebSphere MQ también puede definir el SYSTEM.PROTECTION.ERROR.QUEUE como una cola alias que apunta a otra cola.

Conceptos clave

Conozca los conceptos esenciales de Advanced Message Security para comprender cómo trabaja la herramienta y cómo utilizarla de forma efectiva.

Infraestructura de claves públicas

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 301
Para proporcionar protección de cifrado transparente para las aplicaciones de WebSphere MQ, Advanced Message Security utiliza el archivo de almacén de claves, donde se almacenan certificados de clave pública y una clave privada.

Certificados digitales

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*.

Gestor de autorizaciones sobre objetos

El Gestor de autorizaciones sobre objetos (OAM) es el componente de servicio de autorización que se proporciona con los productos WebSphere MQ.

El acceso a las entidades de Advanced Message Security se controla mediante grupos de usuarios de WebSphere 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:

- El acceso a objetos de Advanced Message Security mediante 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

Tecnología soportada

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 WebSphere MQ:

- Interfaz de cola de mensajes (MQI)
- WebSphere MQ Java Message Service (JMS) 1.0.2 y 1.1.
- Clases base de WebSphere MQ para Java
- Clases de WebSphere 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

Conozca las limitaciones de IBM WebSphere MQ Advanced Message Security.

- Las opciones de IBM WebSphere MQ siguientes no se soportan:
  - Publicación/suscripción
  - Conversión de datos de canal.
  - Listas de distribución.
  - Segmentación de mensajes de aplicación
  - Uso de aplicaciones sin hebras utilizando la salida de API en plataformas HP-UX.
  - IBM WebSphere MQ classe for .NET en una modalidad gestionada (conexiones de cliente o de enlaces).
  - Cliente de Message Service para aplicaciones .NET (XMS).
  - Cliente de Message Service para aplicaciones C/C++ (XMS supportPac IA94).
- Todas las aplicaciones Java dependen de IBM Java Runtime.

  IBM WebSphere MQ Advanced Message Security no es compatible con el JRE proporcionado por otros proveedores.
- JMS y aplicaciones cliente de Java que utilizan IBM WebSphere MQ Advanced Message Security en modalidad de cliente.

  Cualquier JMS, o Java, aplicación cliente (incluidos agentes de IBM WebSphere MQ Explorer y IBM WebSphere MQ Managed File Transfer) no se puede utilizar IBM WebSphere MQ Advanced Message Security en modalidad de cliente con un gestor de colas de WebSphere MQ anterior a Version 7.5.

  Para poder utilizar las políticas de protección de mensajes, estas aplicaciones deben interactuar con un gestor de colas IBM WebSphere MQ Version 7.5, o conectarse en la modalidades de enlaces locales con un gestor de colas en la misma máquina que la aplicación.
- Debe evitar colocar dos o más certificados con los mismos nombres distinguidos, en un único archivo de almacén de claves, porque las funciones del interceptor de IBM WebSphere MQ Advanced Message Security con dichos certificados no están definidas.
- El adaptador de recursos de IBM WebSphere MQ Version 7.5 no da soporte a IBM WebSphere MQ Advanced Message Security. Si es necesario utilizar la protección de mensajes con las aplicaciones IBM WebSphere MQ classes for JMS o IBM WebSphere MQ classes for Java que se ejecutan en un entorno de servidor de aplicaciones, entonces:
  - El servidor de aplicaciones debe configurarse para utilizar el adaptador de recursos de Version 8.0 o posterior.
  - O debe utilizarse la intercepción de agente de canal de mensajes (MCA).

Escenarios de usuario

Conozca los posibles escenarios para comprender cuáles son los objetivos empresariales que se pueden alcanzar con Advanced Message Security.

Guía de inicio rápido para plataformas Windows

Use esta guía para configurar rápidamente IBM 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 Características de IBM WebSphere MQ para sistemas Windows para obtener más detalles.

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 WebSphere MQ adecuados, consulte setmqenv.

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 punto en el que entran en la infraestructura de WebSphere MQ a través de la interfaz WebSphere MQ estándar. La configuración básica se realiza en WebSphere MQ y se define en los pasos siguientes.

Puede utilizar WebSphere 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 \WebSphere 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 pongan 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
```

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 284.

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 JKS y certificados utilizando el mandato **keytool** , que forma parte del JRE (consulte "Guía de inicio rápido para clientes Java" en la página 290 para obtener más detalles). Para todos los demás lenguajes y para las aplicaciones Java que 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` separado. 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 " 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)” en la página 301.
- *%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 301.
- 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/alice/AMS/bobkey.kdb" -pw passw0rd
-label Bob_Cert -target bob_public.arm

runmqakm -cert -add -db "C:/Documents and Settings/bob/AMS/alicekey.kdb" -pw passw0rd -label
Bob_Cert -file bob_public.arm
```

Resultados

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 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

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.

Guía de inicio rápido para plataformas UNIX

Utilice esta guía para configurar rápidamente IBM Advanced Message Security para proporcionar seguridad de mensajes en plataformas 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
- MQ Advanced Message Security

Consulte los temas siguientes para ver los nombres de componentes en cada plataforma específica:

- Componentes deIBM WebSphere MQ para sistemas Linux
- Componentes deIBM WebSphere MQ para sistemas HP-UX
- Componentes deIBM WebSphere MQ para sistemas AIX
- Componentes deIBM WebSphere 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 punto en el que entran en la infraestructura de WebSphere MQ a través de la interfaz WebSphere MQ estándar. La configuración básica se realiza en WebSphere MQ y se define en los pasos siguientes.

Puede utilizar WebSphere 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_INSTALL_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 pongan 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
```

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 290.

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 JKS y certificados utilizando el mandato **keytool** , que forma parte del JRE (consulte “Guía

de inicio rápido para clientes Java” en la página 290 para obtener más detalles). Para todos los demás lenguajes y para las aplicaciones Java que 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)" en la página 301.
- `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 301.

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 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

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.

Guía de inicio rápido para clientes Java

Utilice esta guía para configurar rápidamente IBM Advanced Message Security para proporcionar seguridad de mensajes para aplicaciones Java que se conectan utilizando enlaces de cliente. La guía describe cómo crear un almacén de claves para verificar las identidades de usuario y definir políticas de firma/cifrado para el 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 punto en el que entran en la infraestructura de WebSphere MQ a través de la interfaz WebSphere MQ estándar. La configuración básica se realiza en WebSphere 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** mostrará detalles sobre TEST.Q:

```
DISPLAY Q(TEST.Q)
```

2. Crear y autorizar usuarios

Acerca de esta tarea

En nuestro 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 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, 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
```

3. También debe permitir que los dos usuarios examinen la cola de políticas del sistema y pongan 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
```

Resultados

Se crean los usuarios y se les otorgan las autorizaciones necesarias.

Qué hacer a continuación

Para verificar si los pasos se han llevado a cabo correctamente, utilice los ejemplos `JmsProducer` y `JmsConsumer` tal como se describe en la sección "7. Probar la configuración" en la página 294.

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 utilizando 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 utilizando 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`. Tal vez desee crearlo en el mismo directorio utilizado en la **Guía de inicio rápido** (Windows o UNIX) para su plataforma.

   Nota: Este directorio se conoce como *keystore-dir* en los psos siguientes
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` será 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 será 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 `keystore.conf` porque ha seguido la **Guía de inicio rápido** (Windows o UNIX), puede editar el existente para añadir en las líneas anteriores.
- Para obtener más información, consulte “Estructura del archivo de configuración del almacén de claves (keystore.conf)” en la página 301.

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 IBM GSKit Keyman (ikeyman) realiza una distinción en que *extrae* certificados (claves públicas) y *exporta* 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 WebSphere 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 WebSphere 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 IBM WebSphere 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

A fin de proteger totalmente conexiones de cola remota, se debe establecer misma política en la cola remota y la cola local a la 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 WebSphere MQ para su proceso. Después de procesar el mensaje, Advanced Message Security lo coloca en una cola remota, WebSphere 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 utilizando WebSphere Message Broker

IBM Advanced Message Security puede proteger los mensajes en una infraestructura donde se ha instalado WebSphere Message Broker versión 8.0.0.1 (o posterior). Para ello debe comprender la naturaleza de ambos productos antes de aplicar la seguridad en el entorno de WebSphere Message Broker.

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 implica que para proteger los mensajes que fluyen a través de WebSphere Message Broker, puede permitir que WebSphere Message Broker procese mensajes sin conocer su contenido (Escenario 1) o que sea un usuario autorizado capaz de recibir y enviar mensajes (Escenario 2).

Caso de ejemplo 1 - El intermediario de mensajes no puede ver el contenido del mensaje

Antes de empezar

Debe tener su WebSphere Message Broker 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 WebSphere Message Broker. Advanced Message Security solo lo utilizan *alice*, *bob* y *cecil*. No es necesario instalarlo ni configurarlo para WebSphere Message Broker.

WebSphere Message Broker 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. WebSphere Message Broker 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). WebSphere Message Broker coloca el mensaje direccionado en un alias no protegido. Los destinatarios recuperan el mensaje de una cola de salida protegida donde IBM WebSphere MQ 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`.

[1] cecil's

```
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 WebSphere Message Broker, 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 WebSphere Message Broker.
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. WebSphere Message Broker recupera el mensaje con el formato protegido del alias AIN. WebSphere Message Broker decide adónde direccionar el mensaje examinando la propiedad `routeTo`, la cual no está cifrada, como ocurre con todas las propiedades. WebSphere Message Broker 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 intermediario de mensajes puede ver el contenido del mensaje

Acerca de esta tarea

En este ejemplo, un grupo de usuarios está autorizado para enviar mensajes a WebSphere Message Broker. Otro grupo está autorizado para recibir los mensajes que ha creado WebSphere Message Broker. La transmisión entre los interlocutores y WebSphere Message Broker no puede ser interceptada.

Tenga en cuenta que WebSphere Message Broker 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 WebSphere Message Broker 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 WebSphere Message Broker. 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 WebSphere Message Broker.

Procedimiento

1. Configure *alice*, *bob*, *cecil* y *dave* y el usuario del servicio WebSphere Message Broker, para utilizar 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 los certificados de *alice*, *bob*, *cecil* y *dave* al usuario de servicio de WebSphere Message Broker.

   Para ello, extraiga a archivos externos cada uno de los certificados que identifican *alice*, *bob*, *cecil* y *dave* y, a continuación, añada los certificados extraídos al almacén de claves de WebSphere Message Broker. 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 WebSphere Message Broker 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 WebSphere Message Broker para cifrar los mensajes correctamente. El usuario de servicio de WebSphere Message Broker necesita los certificados de *Alice* y *Bob* para verificar los autores de los mensajes. El usuario de servicio de WebSphere Message Broker necesita los certificados de *Cecil* y *Dave* para cifrar los mensajes destinados a ellos. *cecil* y *dave* necesitan el certificado de usuario de servicio de WebSphere Message Broker para verificar si el mensaje procede de WebSphere Message Broker.
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 de WebSphere Message Broker 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 denominada `OUT` y defina la política de seguridad con el usuario de servicio de WebSphere Message Broker 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 WebSphere Message Broker, 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 WebSphere Message Broker.
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 permitiendo solo que WebSphere Message Broker los lea. WebSphere Message Broker sólo aceptará mensajes de *Alice* y *Bob*, y rechazará los demás. 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 *Cecil* y *Dave* son capaces de leer, los mensajes no firmados por WebSphere Message Broker se rechazarán.

Utilización de IBM WebSphere MQ Advanced Message Security con IBM WebSphere MQ 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 IBM WebSphere MQ Managed File Transfer.

Antes de empezar

Compruebe que tiene el componente Advanced Message Security instalado en la instalación de WebSphere MQ que aloja las colas utilizadas por IBM WebSphere MQ Managed File Transfer que desea proteger.

Si los agentes de IBM WebSphere MQ 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 IBM WebSphere MQ Managed File Transfer, es probable que queden datos confidenciales desprotegidos en las colas de WebSphere 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 IBM WebSphere MQ Managed File Transfer.

En este caso de ejemplo, se supone una topología simple formada por una máquina con dos colas de IBM WebSphere MQ Managed File Transfer y dos agentes, `AGENT1` y `AGENT2`, que comparten un solo gestor de colas, `hubQM`, tal como se describe en el caso de ejemplo Transferencia de archivos básica utilizando los scripts. 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 IBM WebSphere MQ Managed File Transfer. 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 IBM WebSphere MQ 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 IBM WebSphere MQ 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 clientes Java” en la página 290.

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=IBM, L=Hursley, ST=Hampshire, C=GB
   ```

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 IBM WebSphere MQ 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=IBM,
   L=Hursley, ST=Hampshire, C=GB"
   -e AES128 -r "CN=ftagent, OU=MFT, O=IBM, L=Hursley, ST=Hampshire, C=GB"
   ```

3. Asegúrese de que el usuario que ejecuta el proceso de agente de IBM WebSphere MQ Managed File Transfer 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 IBM WebSphere MQ 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.

Instalación de IBM WebSphere MQ Advanced Message Security

Instalar el componnete IBM WebSphere MQ Advanced Message Security en varias plataformas.

Acerca de esta tarea

Si desea ver los procedimientos completos de instalación, consulte Instalación de IBM WebSphere MQ Advanced Message Security .

Tareas relacionadas

DesinstalaciónIBM WebSphere MQ Advanced Message Security

Utilización de almacenes de claves y certificados

Para proporcionar protección de cifrado transparente para las aplicaciones de WebSphere MQ, Advanced Message Security utiliza el archivo de almacén de claves, donde se almacenan certificados de clave pública y una clave privada.

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.

La ubicación del almacén de claves se proporciona en el archivo de configuración del almacén de claves, que es `keystore.conf` de forma predeterminada. 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 el siguiente formato de archivos de almacén de claves: `.kdb`, `.jceks`, `.jks`.

La ubicación predeterminada del archivo `keystore.conf` es:

- En plataformas UNIX : `$HOME/.mqs/keystore.conf`
- En las plataformas Windows: *%HOMEDRIVE%%HOMEPATH%*`\.mqs\keystore.conf`

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*

    Nota: La vía de acceso en Windows puede y debe especificar la letra de unidad si hay más de una letra de unidad disponible.

Conceptos relacionados

“Nombres distinguidos de emisor” en la página 315
Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola.

“Nombres distinguidos de destinatario” en la página 316
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)

El archivo de configuración de almacén de claves (`keystore.conf`) apunta Advanced Message Security a la ubicación del almacén de claves adecuado.

Hay dos tipos de configuración CMS y Java (JKS y JCEKS). A las entradas de configuración de CMS se les añade el prefijo `cms.` y a Java se les añade el prefijo `jks.` o `jceks.` en función del tipo de almacén de claves.

El archivo de configuración, según el tipo de archivo de configuración, puede tener una de las estructuras siguientes:

```
cms.keystore = /<dir>/<keystore_file>
cms.certificate = certificate_label

jceks.keystore = <dir>/Keystore
jceks.certificate = <certificate_label>
jceks.encrypted = no
jceks.keystore_pass = <password>
jceks.key_pass = <password>
jceks.provider = IBMJCE
```

```
jks.keystore = <dir>/Keystore
jks.certificate = <certificate_label>
jks.encrypted = no
jks.keystore_pass = <password>
jks.key_pass = <password>
jks.provider = IBMJCE
```

Los parámetros del archivo de configuración se definen del modo siguiente:

keystore
Vía de acceso del archivo de almacén de claves.

Importante:

- La vía de acceso del archivo de almacén de claves no debe incluir la extensión de archivo.
- Para archivos de almacén de claves Java, IBM WebSphere MQ AMS da soporte a los siguientes formatos de archivo: `.jks`, `.jceks`, `.jck`.

certificate
Etiqueta del certificado.

encrypted
Estado de la contraseña.

keystore_pass
Contraseña del archivo de almacén de claves.

Nota:

- Para el almacén de claves de CMS, IBM WebSphere MQ 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.
- El almacenamiento de contraseñas como texto sin cifrar supone un riesgo de seguridad.

key_pass
La contraseña de la clave privada del usuario.

Importante: El almacenamiento de contraseñas como texto sin cifrar pueden suponer un riesgo de seguridad.

provider
El proveedor de seguridad Java que implementa algoritmos criptográficos necesarios para el certificado de almacén de claves.

Nota: Actualmente IBMJCE es el único proveedor que se puede utilizar con Advanced Message Security.

Importante: La información almacenada en el almacén de claves es esencial para el flujo seguro de los datos enviados utilizando WebSphere MQ, razón por la cual los administradores de seguridad deben prestar especial atención al asignar permisos de archivo para estos archivos.

Esto es un ejemplo del archivo `keystore.conf`:

```
cms.keystore = c:\Documents and Settings\Alice\AliceKeystore
cms.certificate = AliceCert

jceks.keystore = c:/Documents and Settings/Alice/AliceKeystore
jceks.certificate = AliceCert
jceks.encrypted = no
jceks.keystore_pass = <password>
jceks.key_pass = <password>
jceks.provider = IBMJCE
```

Tareas relacionadas

“Protección de contraseñas en Java” en la página 312
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.

Interceptación del agente de canal de mensajes (MCA)

La intercepción de MCA permite a un gestor de colas que se ejecuta en IBM WebSphere MQ habilitar de forma selectiva las políticas que se van a aplicar para los canales de conexión de servidor.

La interceptación de MCA permite a los clientes que permanecen fuera de IBM WebSphere MQ AMS seguir conectados a un gestor de colas y cifrar y descifrar sus mensajes.

La intercepción MCA se ha diseñado para proporcionar la prestación IBM WebSphere MQ AMS cuando IBM WebSphere MQ AMS no se puede habilitar en el cliente. Tenga en cuenta que utilizar la intercepción MCA y un cliente habilitado para IBM WebSphere MQ AMS lleva a una doble protección que podría ser problemática para recibir aplicaciones.

Si se notifica un error `2085 (MQRC_UNKNOWN_OBJECT_NAME)` si está utilizando un cliente Version 7.5 o posterior para conectarse a un gestor de colas desde una versión anterior del producto, debe inhabilitar IBM WebSphere MQ Advanced Message Security en el cliente. Para obtener más información, consulte “Inhabilitación de IBM WebSphere MQ Advanced Message Security en el cliente” en la página 305.

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 IBM WebSphere MQ AMS, consulte “Utilización de almacenes de claves y certificados” en la página 301.

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 de MCA, solo se puede utilizar un tipo de almacén de claves CMS.

Para ver un ejemplo de la configuración de la interceptación de MCA, consulte “Ejemplo de interceptación de MCA de IBM WebSphere MQ AMS” en la página 303.

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.

Ejemplo de interceptación de MCA de IBM WebSphere MQ AMS

Una tarea de ejemplo de cómo configurar una interceptación de MCA de IBM WebSphere MQ 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.

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 Version 7.5, IBM WebSphere MQ AMS era un producto complementario que se tenía que instalar por separado e interceptores configurados para proteger aplicaciones. Desde Version 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 ha utilizado un Version 7.5 o un cliente posterior, esto haría que el mensaje se protegiera dos veces, porque el interceptor de tiempo de ejecución del cliente MQ y el interceptor MCA protegerían ambos el mensaje ya que viene 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/mq75
KEYSTORELOC=/home/testusr/ssl/ams1
mkdir -p $KEYSTORELOC
chmod -R 777 $KEYSTORELOC
chown -R mqm:mqm $KEYSTORELOC
export PATH=$PATH:$INSTLOC/gskit8/bin
echo "PATH = $PATH"
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$INSTLOC/gskit8/lib64

gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd -stash
gsk8capicmd_64 -keydb -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd -stash
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/alicekey.kdb -pw passw0rd
-label alice_cert -dn "cn=alice,O=IBM,c=IN" -default_cert yes
gsk8capicmd_64 -cert -create -db $KEYSTORELOC/bobkey.kdb -pw passw0rd
-label bob_cert  -dn "cn=bob,O=IBM,c=IN"  -default_cert yes
```

2. 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 Version 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 WebSphere MQ Version 7.5:

```
/opt/mq75/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 WebSphere MQ Version 7.5:

```
/opt/mqm/samp/bin/amqsgetc TESTQ TESTQMGR
```

Tareas relacionadas

"Guía de inicio rápido para clientes Java" en la página 290
Utilice esta guía para configurar rápidamente IBM Advanced Message Security para proporcionar seguridad de mensajes para aplicaciones Java que se conectan utilizando enlaces de cliente. La guía describe cómo crear un almacén de claves para verificar las identidades de usuario y definir políticas de firma/cifrado para el gestor de colas.

Referencia relacionada

"Limitaciones conocidas" en la página 279
Conozca las limitaciones de IBM WebSphere MQ Advanced Message Security.

Inhabilitación de IBM WebSphere MQ Advanced Message Security en el cliente

Debe inhabilitar IBM WebSphere MQ Advanced Message Security (AMS) en el cliente si está utilizando un cliente Version 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 Version 7.5, IBM WebSphere MQ Advanced Message Security (AMS) se habilita automáticamente en un cliente IBM WebSphere MQ , por lo que, de forma predeterminada, el cliente intenta comprobar las políticas de seguridad para los objetos en el gestor de colas. Sin embargo, los servidores de versiones anteriores del producto, por ejemplo Version 7.1, no tienen AMS habilitado, lo que hace que se notifique un 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 en el cliente de la forma siguiente:

- Para clientes de Java, en cualquiera de las formas siguientes:
  - V7.5.0.4 Estableciendo una variable de entorno AMQ_DISABLE_CLIENT_AMS.
  - V7.5.0.4 Estableciendo la propiedad del sistema Java com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS.
  - V7.5.0.5 Utilizando la propiedad DisableClientAMS, bajo la stanza **Security** en el archivo mqclient.ini.
- Para clientes C, de cualquiera de una de las formas siguientes:

– V 7.5.0.4 Estableciendo una variable de entorno AMQ_DISABLE_CLIENT_AMS.

– V 7.5.0.5 Utilizando la propiedad DisableClientAMS, bajo la stanza **Security** en el archivo `mqclient.ini`.

Procedimiento

- Para inhabilitar AMS en el cliente, utilice una de las opciones siguientes:

  V 7.5.0.4 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 un cliente Version 7.5, o posterior, IBM WebSphere MQ classes for Java o IBM WebSphere MQ classes for JMS.

  También puede utilizar AMQ_DISABLE_CLIENT_AMS para inhabilitar la funcionalidad de AMS para clientes C.

  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
  ```

  V 7.5.0.4 Propiedad del sistema com.ibm.mq.cfg.AMQ_DISABLE_CLIENT_AMS

  Para los clientes IBM WebSphere MQ classes for JMS y IBM WebSphere MQ classes for Java , establezca 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.

  V 7.5.0.5 Propiedad `DisableClientAMS` en el archivo `mqclient.ini`

  Para clientes IBM WebSphere MQ classes for JMS y IBM WebSphere MQ classes for Java , y para clientes C, añada el nombre de propiedad `DisableClientAMS` bajo la stanza **Security** del 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

Si desea más información sobre problemas con la apertura de colas protegidas de AMS, consulte “Problema al abrir colas protegidas utilizando JMS” en la página 330.

Conceptos relacionados

“Interceptación del agente de canal de mensajes (MCA)” en la página 303
La intercepción de MCA permite a un gestor de colas que se ejecuta en IBM WebSphere MQ habilitar de forma selectiva las políticas que se van a aplicar para los canales de conexión de servidor.

Tareas relacionadas

Configuración de un cliente utilizando un archivo de configuración

Referencia relacionada

El archivo de configuración IBM WebSphere 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 WebSphere MQ” en la página 36.

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 debe establecerse así: para los certificados del estándar X.509 V3 o posterior que se utilizan para la calidad de protección de integridad, si se establecen las extensiones de uso de clave, deben incluir al menos uno de estos dos:

- **`nonRepudiation`**
- **`digitalSignature`**

Para la calidad de protección de privacidad, si las extensiones de uso de clave están definidas, deben también incluir la extensión **`keyEncipherment`**.

Conceptos relacionados

“Calidad de protección” en la página 317

Las políticas de protección de datos de Advanced Message Security implican una calidad de protección (QOP).

Métodos de validación de certificados en IBM WebSphere MQ Advanced Message Security

Puede utilizar IBM WebSphere MQ 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.

IBM WebSphere MQ AMS le permite verificar la validez de un certificado utilizando Online Certificate Status Protocol (OCSP) o la lista de revocación de certificados (CRL).

IBM WebSphere MQ AMS se puede configurar para realizar la comprobación mediante OCSP o CRL, o por ambos métodos. Si se habilitan ambos métodos, entonces IBM WebSphere MQ 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, IBM WebSphere MQ AMS utiliza la comprobación por CRL.

Conceptos relacionados

“Protocolo de estado de certificado en línea (OCSP)” en la página 307

Online Certificate Status Protocol (OCSP) determina si un certificado se ha revocado, y, por lo tanto, ayuda a determinar si el certificado es fiable.

“Listas de revocación de certificados (CRL)” en la página 309

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)

Online Certificate Status Protocol (OCSP) determina si un certificado se ha revocado, y, por lo tanto, ayuda a determinar si el certificado es fiable.

Habilitación de la comprobación de OCSP en interceptores nativos
Para habilitar la incorporación de OCSP (Online Certificate Status Protocol) en Advanced Message Security, debe modificar el archivo de configuración del almacén de claves.

Procedimiento

Añada las opciones siguientes al archivo de configuración del almacén de claves:

Nota: Los valores proporcionados en la tabla para cada opción son los valores predeterminados.

Debe especificar uno de los valores siguientes:

- `ocsp.enable=on`
- `ocsp.url=<`*`reposnder_URL`*`>`
- `ocsp.http.proxy.host=<`*`OCSP_proxy`*`>`

Opción	Descripción
`ocsp.enable=off`	Habilite la comprobación de OCSP si el certificado sometido a comprobación tiene una extensión de AIA (Authority Info Access) con un método de acceso PKIX_AD_OCSP que contiene un URI que apunta a la ubicación del respondedor de OCSP. Valores posibles: on/`off`.
`ocsp.url=<`*`reposnder_URL`*`>`	Dirección de URL del respondedor de OCSP.
`ocsp.http.proxy.host=<`*`OCSP_proxy`*`>`	Dirección de URL del servidor proxy de OCSP.
`ocsp.http.proxy.port=<`*`port_number`*`>`	Número de puerto del servidor proxy de OCSP.
`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.
`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.

Habilitación de la comprobación de OCSP en Java
Para habilitar la incorporación 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 tiene configurado Authority Information Access (AIA).

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 utiliza Java Security Manager, también debe completar la configuración, añada el siguiente permiso 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 siguientes permisos 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)

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.IBM WebSphere MQ AMS verifica la vía de acceso del certificado utilizando un algoritmo de validación PKIX. Cuando se crea y verifica la cadena, IBM WebSphere MQ 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, IBM WebSphere MQ 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, IBM WebSphere MQ 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. IBM WebSphere MQ 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, IBM WebSphere MQ AMS no realiza ninguna comprobación de validez por CRL. Para cada certificado de entidad emisora de certificados (certificados de CA), IBM WebSphere MQ 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).

Procedimiento

Añada las opciones siguientes al archivo de configuración:

Nota: Los valores proporcionados en la tabla para cada opción son los valores predeterminados.

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 Java de IBM WebSphere MQ AMS se conecta correctamente a un servidor LDAP, no intenta descargar las CRL de los servidores restantes proporcionados.
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*>	Conexión al 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 IBM WebSphere MQ AMS concluye.
`crl.http.cache_size=0`	Tamaño de la memoria caché de HTTP, en bytes.

Habilitación del soporte de lista de revocación de certificados en Java
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 IBM WebSphere MQ 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.
`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 Java de IBM WebSphere MQ AMS se conecta correctamente a un servidor LDAP, no intenta descargar las CRL de los servidores restantes proporcionados. 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.
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.

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 301.

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
```

Administración de políticas de seguridad de IBM WebSphere MQ Advanced Message Security

IBM WebSphere MQ 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

Las políticas de seguridad de IBM Advanced Message Security son objetos conceptuales que describen la forma en que se cifra y firma criptográficamente un mensaje.

Para obtener más detalles sobre los atributos de política de seguridad, consulte los temas subordinados siguientes:

Conceptos relacionados

"Calidad de protección" en la página 317
Las políticas de protección de datos de Advanced Message Security implican una calidad de protección (QOP).

"Atributos de política de seguridad" en la página 317
Puede utilizar Advanced Message Security para seleccionar un algoritmo o método determinados para proteger los datos.

Nombre de política

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 de uno a uno entre una política de Advanced Message Security (IBM WebSphere MQ 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 IBM WebSphere MQ 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

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

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

El atributo de tolerancia indica si IBM 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

Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola.

IBM Advanced Message Security (IBM WebSphere MQ AMS) no comprueba si un usuario válido ha colocado un mensaje en una cola protegida por datos hasta que se recupera el mensaje. En este momento, si la política estipula uno o varios emisores válidos y el usuario que colocó el mensaje en la cola no está en la lista de emisores válidos, IBM WebSphere MQ AMS devuelve un error a la aplicación y coloca el mensaje en su cola de errores.

Una política puede tener 0 o más nombres distinguidos de emisor válidos. Si no se especifica ningún nombre distinguido de emisor para la política, cualquier usuario puede colocar mensajes con datos protegidos en la cola, siempre y cuando el certificado del usuario sea de confianza.

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.
DE	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.
- IBM WebSphere MQ AMS permite utilizar nombres distinguidos con caracteres pertenecientes solamente al conjunto de caracteres Latin-1 Para crear nombres distinguidos con caracteres de ese conjunto, debe primero crear un certificado con un nombre distinguido que se crea en la codificación UTF-8 utilizando plataformas UNIX con la codificación UTF-8 activada o mediante el programa de utilidad iKeyman. A continuación, debe crear una política desde una plataforma UNIX con la codificación UTF-8 activada o utilizar el conector de IBM WebSphere MQ AMS con WebSphere MQ.

Conceptos relacionados

"Nombres distinguidos de destinatario" en la página 316

Los nombres distinguidos de destinatario identifican a usuarios que están autorizados a recuperar mensajes de una cola.

Nombres distinguidos de destinatario

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.
DE	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 nombres distinguidos con caracteres de ese conjunto, debe primero crear un certificado con un nombre distinguido que se crea en la codificación UTF-8 utilizando plataformas UNIX con la codificación UTF-8 activada o mediante el programa de utilidad iKeyman. 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 WebSphere MQ.

Conceptos relacionados

"Nombres distinguidos de emisor" en la página 315

Los nombres distinguidos de emisor identifican usuarios que están autorizados a colocar mensajes en una cola.

Atributos de política de seguridad

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. La tabla siguiente muestra los atributos de política de seguridad existentes en Advanced Message Security:

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 existentes en Advanced Message Security dependen de algoritmos de cifrado 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.
- 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 no estipula un algoritmo de firma ni un algoritmo de cifrado tiene una calidad de protección NONE. Advanced Message Security no proporciona protección de datos para las colas que tienen una política con un QOP de 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.

Todas las tareas administrativas relacionadas con las políticas de seguridad se ejecutan desde la ubicación siguiente:

- En plataformas UNIX : `<MQInstallRoot>/bin`
- En las plataformas Windows, las tareas administrativas se pueden ejecutar desde cualquier ubicación porque la variable de entorno PATH se actualiza durante la instalación.

Tareas relacionadas

“Creación de políticas de seguridad” en la página 318
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 la página 319
Puede utilizar Advanced Message Security para modificar los detalles de las políticas de seguridad que ya ha definido.

“Visualización y volcado de políticas de seguridad” en la página 320
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 la página 321
Para eliminar las políticas de seguridad en Advanced Message Security, debe utilizar el mandato `setmqspl` .

Creación de políticas de seguridad

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 WebSphere MQ.
- Debe tener autorizaciones de `+connect +inq +chg` necesarias para crear una política de seguridad. Para conocer la sintaxis completa del mandato de cambio de autorización, consulte **`setmqaut`** .
- Asegúrese de que tiene los permisos necesarios para operar en colas y gestores de colas de WebSphere MQ. Para obtener más información, consulte “Cómo otorgar permisos OAM” en la página 323.

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 firman mediante el algoritmo SHA1 y se cifran utilizando el algoritmo AES256 para los certificados con el nombre distinguido CN=joe,O=IBM,C=US y el nombre distinguido: CN=jane,O=IBM,C=US. Esta política está asociada a MY.QUEUE:

```
$ setmqspl -m QMGR -p MY.QUEUE -s SHA1 -e AES256 -r CN=joe,O=IBM,C=US -r CN=jane,O=IBM,C=US
```

A continuación se muestra un ejemplo de creación de una política en el gestor de colas QMGR. La política especifica que los mensajes se cifran utilizando el algoritmo DES para los certificados con los nombres distinguidos: CN=john,O=IBM,C=US y CN=jeff,O=IBM,C=US, y se firman con el algoritmo MD5 para el certificado con el nombre distinguido: CN=phil,O=IBM,C=US

```
$ setmqspl -m QMGR -p MY.OTHER.QUEUE -s MD5 -e DES -r CN=john,O=IBM,C=US -r CN=jeff,O=IBM,C=US
-a CN=phil,O=IBM,C=US
```

Nota:

- 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

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 las autorizaciones de `+connect +inq +chg` necesarias para crear políticas de seguridad. Para conocer la sintaxis completa del mandato de cambio de autorización, consulte **setmqaut**.

Acerca de esta tarea

Para cambiar las políticas de seguridad, aplique el mandato `setmqspl` a una política ya existente proporcionando nuevos atributos.

Ejemplo

El ejemplo siguiente crea una política llamada MYQUEUE en un gestor de colas llamado QMGR, y especifica que los mensajes se cifrarán utilizando el algoritmo RC2 para los certificados con el nombre distinguido: CN=bob,O=IBM,C=US, y se firmarán con el algoritmo SHA1 para los certificados con el nombre distinguido: CN=jeff,O=IBM,C=US.

```
setmqspl -m QMGR -p MYQUEUE -e RC2 -s SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Para 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 SHA1 -a CN=jeff,O=IBM,C=US -r CN=alice,O=IBM,C=US
```

Referencia relacionada

setmqspl

Visualización y volcado de políticas de seguridad

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 los permisos +connect +inq +dsp necesarios aplicados a un gestor de colas para visualizar y volcar las políticas de seguridad. Para conocer la sintaxis completa del mandato de cambio de autorización, consulte **setmqaut** .

Acerca de esta tarea

A continuación se muestra una lista de los distintivos del mandato dspmqspl:

Tabla 27. 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 este ejemplo crearemos dos políticas de seguridad para venus.queue.manager:

```
setmqspl -m venus.queue.manager -p AMS_POL_04_ONE -s MD5 -a "CN=signer1,O=IBM,C=US" -e NONE
setmqspl -m venus.queue.manager -p AMS_POL_06_THREE -s MD5 -a "CN=another signer,O=IBM,C=US" -e
NONE
```

Este 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: MD5
Encryption algorithm: NONE
Signer DNs:
  CN=signer1,O=IBM,C=US
Recipient DNs: -
Toleration: 0
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Policy Details:
Policy name: AMS_POL_06_THREE
Quality of protection: INTEGRITY
Signature algorithm: MD5
Encryption algorithm: NONE
Signer DNs:
  CN=another signer,O=IBM,C=US
Recipient DNs: -
Toleration: 0
```

Este 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: MD5
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 MD5 -a "CN=signer1,O=IBM,C=US" -e NONE

dspmqspl -m venus.queue.manager -export > policies.[bat|sh]
```

Para importar una política de seguridad:

- En las plataformas Windows, ejecute `policies.bat`
- En plataformas UNIX:
  1. Inicie sesión como un usuario que pertenece al grupo de administración `mqm` WebSphere MQ.
  2. Emita `. policies.sh`.

Referencia relacionada

Lista completa de los atributos del mandato dspmqspl

Eliminación de políticas de seguridad

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 las autorizaciones de `+connect +inq +chg` necesarias para crear políticas de seguridad. Para conocer la sintaxis completa del mandato de cambio de autorización, consulte **`setmqaut`** .

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

Las colas del sistema permiten la comunicación entre WebSphere 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 WebSphere 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 la página 318.

Para utilizar la protección de colas del sistema en las plataformas Windows, copie el archivo `keystore.conf` en el directorio siguiente:

```
c:\Documents and Settings\Default User\.mqs\keystore.conf
```

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, los mensajes de mandato que no pasan la comprobación de nombre distinguido de firmante (DN) no son ejecutados por el servidor de mandatos y no se envían a la cola de manejo de errores de Advanced Message Security. Los mensajes que se envían como respuestas a colas dinámicas temporales de WebSphere MQ Explorer no son protegidos por WebSphere MQ AMS.

Los cambios realizados en políticas de seguridad de Advanced Message Security requieren reiniciar el servidor de mandatos de WebSphere MQ.

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
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.LOGGER.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.PUBSUB.EVENT
- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.STATISTICS.QUEUE
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE
- SYSTEM.AUTH.DATA.QUEUE
- SYSTEM.BROKER.ADMIN.STREAM
- SYSTEM.BROKER.CONTROL.QUEUE
- SYSTEM.BROKER.DEFAULT.STREAM
- SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS
- SYSTEM.CHANNEL.INITQ
- SYSTEM.CHANNEL.SYNCQ
- SYSTEM.CICS.INITIATION.QUEUE
- SYSTEM.CLUSTER.COMMAND.QUEUE
- SYSTEM.CLUSTER.HISTORY.QUEUE
- SYSTEM.CLUSTER.REPOSITORY.QUEUE
- SYSTEM.CLUSTER.TRANSMIT.QUEUE
- SYSTEM.DEAD.LETTER.QUEUE
- SYSTEM.DURABLE.SUBSCRIBER.QUEUE
- SYSTEM.HIERARCHY.STATE
- SYSTEM.INTER.QMGR.CONTROL
- SYSTEM.INTER.QMGR.FANREQ
- SYSTEM.INTER.QMGR.PUBS

- SYSTEM.INTERNAL.REPLY.QUEUE
- SYSTEM.PENDING.DATA.QUEUE
- SYSTEM.PROTECTION.ERROR.QUEUE
- SYSTEM.PROTECTION.POLICY.QUEUE
- SYSTEM.RETAINED.PUB.QUEUE
- SYSTEM.SELECTION.EVALUATION.QUEUE
- SYSTEM.SELECTION.VALIDATION.QUEUE

Cómo otorgar permisos OAM

Los permisos de archivos autorizan a todos los usuarios ejecutar los mandatos `setmqspl` y `dspmqspl`. Sin embargo, IBM 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 WebSphere 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
```

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 mensajes para sucesos de mandato y de configuración generados por WebSphere MQ son mensajes con formato PCF que se envían a colas dedicadas.

Los mensajes para sucesos de configuración se envían a la cola SYSTEM.ADMIN.CONFIG.EVENT en el gestor de colas donde se produce el suceso.

Los mensajes para sucesos de mandato se envían a la cola SYSTEM.ADMIN.COMMAND.EVENT en el gestor de colas donde se produce el suceso.

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 la página 318, que produce dos mensajes de suceso de WebSphere MQ:
  - Un suceso de configuración
  - Un suceso de mandato
- “Modificación de políticas de seguridad” en la página 319, que produce tres mensajes de suceso de WebSphere 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 políticas de seguridad” en la página 320, que produce un solo mensajes de suceso de WebSphere MQ:
  - Un suceso de mandato

- "Eliminación de políticas de seguridad" en la página 321, que produce dos mensajes de suceso de WebSphere 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 mandato, establezca CMDEV en ENABLED. Para habilitar sucesos de mandato excepto para los mandatos MQSC DISPLAY y los mandatos PCF Inquire, 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 Websphere 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 WebSphere 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.

Command

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 valores se definen en WebSphere MQ 7.5 `cmqcfc.h`

Se devuelve: Siempre.

CommandData contiene elementos PCF que componen el mandato PCF.

Formato del mensaje de suceso de configuración

Los sucesos de configuración son mensajes PCF de formato Advanced Message Security estándar.

Para obtener los valores posibles para el descriptor de mensaje MQMD, consulte 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 mensajes consta de la estructura MQCFH y la estructura de parámetros que le sigue. Para ver los posibles valores de MQCFH, consulte 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 WebSphere 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 WebSphere MQ 7.5 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 WebSphere MQ 7.5 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 WebSphere MQ 7.5 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 WebSphere MQ 7.5 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 WebSphere MQ 7.5 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 WebSphere MQ 7.5 o en el archivo `cmqc.h`.
Se devuelve:	Siempre que hay un algoritmo de cifrado definido en la política de WebSphere MQ

SignerDNs

Descripción:	Nombre distinguido de los firmantes permitidos.
Identificador:	**MQCA_SIGNER_DN**
Tipo de datos:	MQCFSL
Valor:	**2113** - valor numérico definido en WebSphere MQ 7.5 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 WebSphere MQ.

RecipientDNs

Descripción:	Nombre distinguido de los firmantes permitidos.
Identificador:	**MQCA_RECIPIENT_DN**
Tipo de datos:	MQCFSL
Valor:	**2114** - valor numérico definido en WebSphere MQ 7.5 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 WebSphere MQ.

Problemas y soluciones

Esta sección muestra cómo solucionar los problemas que pueden surgir con una instalación de IBM. Utilice esta información para identificar y resolver los problemas relacionados con Advanced Message Security.

com.ibm.security.pkcsutil.PKCSException: Error al cifrar el contenido

El error `com.ibm.security.pkcsutil.PKCSException: Error encrypting contents` sugiere que IBM Advanced Message Security tiene problemas con el acceso a algoritmos criptográficos.

Advanced Message Security devuelve el siguiente error:

```
DRQJP0103E   The IBM WebSphere MQ Advanced Message Security Java interceptor failed to protect
message.
  com.ibm.security.pkcsutil.PKCSException: Error encrypting contents
(java.security.InvalidKeyException: Illegal key size or default parameters)
```

Verifique si la política de seguridad JCE en `JAVA_HOME/lib/security/local_policy.jar/*.policy` otorga acceso a los algoritmos de firma utilizados en la política de MQ AMS.

Si el algoritmo de firma que desea utilizar no está especificado en la política de seguridad actual, descargue el archivo de política Java correcto desde las ubicaciones siguientes:

- IBM para Java 1.4.2.

- IBM para Java 5.0.
- IBM para Java 6.0.
- IBM para Java 7.0.

Soporte de OSGi

Para utilizar el paquete OSGi con IBM Advanced Message Security, se requieren parámetros adicionales.

Ejecute el parámetro siguiente durante el inicio del paquete OSGi:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7
```

Cuando se utiliza una contraseña cifrada en keystore.conf, debe añadirse la sentencia siguiente cuando se está ejecutando el paquete OSGi:

```
-Dorg.osgi.framework.system.packages.extra=com.ibm.security.pkcs7,com.ibm.misc
```

Restricción: IBM WebSphere MQ AMS da soporte a la comunicación utilizando sólo MQ Base Java Classes para las colas protegidas desde el paquete OSGi.

Problema al abrir colas protegidas utilizando JMS

Pueden surgir varios problemas al abrir colas protegidas cuando se utiliza IBM WebSphere MQ Advanced Message Security.

Está ejecutando JMS y recibe el error 2085 (MQRC_UNKNOWN_OBJECT_NAME) junto con el error JMSMQ2008.

Ha verificado que ha configurado IBM WebSphere MQ Advanced Message Security tal como se describe en “Guía de inicio rápido para clientes Java” en la página 290.

Una causa posible es que esté utilizando un entorno de ejecución (JRE) que no es de IBMJava. Esta es una limitación conocida que se describe en “Limitaciones conocidas” en la página 279.

No ha establecido la variable de entorno AMQ_DISABLE_CLIENT_AMS.

Resolución del problema

Existen cuatro opciones para evitar este problema:

1. Inicie la aplicación JMS bajo un IBM Java Runtime Environment (JRE) soportado.
2. Mueva la aplicación a la misma máquina donde se está ejecutando el gestor de colas y conéctelo utilizando una conexión en modalidad de enlaces.

   Una conexión en modalidad de enlaces utiliza bibliotecas nativas de la plataforma para realizar las llamadas de API de IBM WebSphere MQ. En consecuencia, el interceptor AMS nativo se utiliza para realizar las operaciones AMS y no hay ninguna dependencia de las posibilidades del JRE.
3. Utilice un interceptor MCA porque esto permite la firma y el cifrado de mensajes tan pronto llegan al gestor de colas, sin que sea necesario que el cliente lleve a cabo ningún proceso de AMS.

   Dado que la protección se aplica al gestor de colas, se debe utilizar un mecanismo alternativo para proteger los mensajes en tránsito del cliente al gestor de colas. Generalmente esto se logra configurando el cifrado SSL/TLS en el canal de conexión del servidor utilizado por la aplicación.
4. Establezca la variable de entorno AMQ_DISABLE_CLIENT_AMS si no desea utilizar IBM WebSphere MQ Advanced Message Security.

En “Interceptación del agente de canal de mensajes (MCA)” en la página 303 encontrará más información.

Nota: Debe haber una política de seguridad para cada cola a la que el interceptor MCA entregará mensajes. En otras palabras, la cola de destino necesita tener una política de seguridad AMS con el nombre distinguido (DN) del firmante y el destinatario que coincida con el del certificado asignado

al interceptor MCA. Es decir, el nombre distinguido del certificado designado por la propiedad `cms.certificate.channel.SYSTEM.DEF.SVRCONN` en `keystore.conf` utilizado por el gestor de colas.

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 IBM 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: